Journal of Uncertain Systems
Vol.4, No.3, pp.171-180, 2010
Online at: www.jus.org.uk

# Eigen Fuzzy Sets and a Genetic Algorithm for Image Information Reconstruction

Ferdinando Di Martino[1,2], Salvatore Sessa[1,*]

[1] *Università degli Studi di Napoli Federico II, Dipartimento di Costruzioni e Metodi Matematici in Architettura, Via Monteoliveto 3, 80134 Napoli, Italy*

[2] *Università degli Studi di Salerno Dipartimento di Matematica ed Informatica 84081 Fisciano (Salerno), Italy*

Received 2 July 2009; Revised 11 September 2009

## Abstract

By normalizing the values of its pixels with respect to the length of the gray scale used, a monochromatic image is interpreted as a fuzzy relation R. We find the GEFS (resp. SEFS) of R, that is the greatest (resp. smallest) eigen fuzzy set with respect to the max-min (resp. min-max) operator. The reconstruction of R is achieved via a genetic algorithm whose initial population of chromosomes is formed by random gray images. The fitness function is based on the GEFS and SEFS of R and of each image-chromosome: the reconstructed image is the image-chromosome with the highest fitness value. In our tests we have used 350 images of sizes 256 × 256 extracted from the SIDBA image dataset consisting of 10000 images (www.cs.cmu.edu/~cil/ vision.html).

**Keywords:** eigen fuzzy set, genetic algorithm, gradient method, image reconstruction

## 1 Introduction

It is well known in literature the usage of fuzzy relation calculus for image processing [3, 4, 5, 8, 9, 10, 11]. Indeed a monochromatic image R of sizes N×N can be interpreted as a fuzzy relation $R: (x,y)\in\{1,\ldots,N\}\times\{1,\ldots,N\}\rightarrow[0,1]$, $R(x,y)=P(x,y)/L$ being the normalized value of the pixel $P(x,y)$ with respect to the length L of the gray scale used.

The concept of eigen fuzzy set has been widely dealt in literature [1, 2, 5, 13, 14, 17, 18]. The greatest eigen fuzzy set (for short, GEFS) of R with respect to the max-min operator [1, 15, 16] and the smallest eigen fuzzy set (for short, SEFS) of R with respect to the min-max operator have been applied to problems of image information retrieval [3] and image reconstruction [8, 9].

It is known the usage of genetic algorithms for fuzzy coding [14]. We reconstruct a monochromatic image of sizes N × N by using its GEFS and SEFS. In [8] the authors adopt essentially two methods: the first one use a convex combination of max-min and min-max operators and the second one uses permutation fuzzy matrices. The authors prove that the first method is more efficient than the second one and a high number of eigen fuzzy sets obtained after many iterations are involved as well. In our study we reconstruct a monochromatic image of sizes N×N by using a genetic algorithm whose chromosomes are random monochromatic images of sizes N×N. We calculate the GEFS and SEFS of the original image and of each image-chromosome, afterwards we define the fitness function as the inverse of the sum of the quadratic differences of the membership values of both GEFS and SEFS and clearly we choose image-chromosome with the greatest fitness value.

This paper is organized as follows: in Section 2 we show how to calculate GEFS and SEFS of a square fuzzy relation, in Section 3 we describe the genetic algorithm used in the image reconstruction, in Section 4 we describe the main tool of [8], in Section 5 we discuss the results on two sample gray images of 256×256 sizes and a comparison with the method of Section 4 is also presented. Section 6 contains concluding comments.

[*] Corresponding author. Email: sessa@unina.it (S. Sessa); Tel. ++390812538907, fax ++390812538909.

## 2 Eigen Fuzzy Sets of a Fuzzy Relation

Let R be a fuzzy relation defined on a finite referential set X, that is $R \in F(X \times X) = \{S : X\times X \rightarrow [0,1]\}$, and A be a fuzzy set of X, that is $A \in F(X)=\{B:X\rightarrow [0,1]\}$, such that

$$R \text{ o } A = A \tag{1}$$

where "o" stands for the max-min operator and A is said an eigen fuzzy set of R with respect to such operator. In terms of membership functions, the Equation (1) is read as

$$A(y) = \max_{x\in X} \{\min\{A(x),R(x,y)\}\} \tag{2}$$

for all $x,y\in X$. Let $A_i \in F(X)$, i=1,2,… be defined recursively by

$$A_1(z) = \max_{x\in X} R(x,z) \ \ \forall z\in X, \ \ A_2= R \text{ o } A_1,\ldots, A_{n+1} = R \text{ o } A_n,\ldots. \tag{3}$$

In accordance to the known literature (e.g., [1], [15], [16]), then there exists an integer $p \in \{1,\ldots,\text{card}X\}$ such that $A_{p+1} = R \text{ o } A_p = A_p$, further $A_p(x) \geq A(x)$ for all $x\in X$ and $A\in F(X)$ satisfying Equation (1), that is $A_p$ is the GEFS of R. We can also consider the following equation, dual of (1), defined as

$$R \bullet A = A \tag{4}$$

where "•" denotes the min-max operator. By dualizing the Equation (2), in terms of membership functions, the Equation (4) is read as

$$A(y) = \min_{x\in X} \{\max\{A(x),R(x,y)\}\} \tag{5}$$

for all $x,y\in X$ and A is defined to be an eigen fuzzy set of R with respect to the min-max operator. Let $B_i\in F(X)$, i=1,2,… be defined recursively by

$$B_1(z) = \min_{x\in X} R(x,z) \ \ \forall z\in X, \ \ B_2= R \bullet B_1,\ldots, B_{n+1} = R \bullet B_n,\ldots. \tag{6}$$

By duality, it is easily seen that there exists an integer $q \in \{1,\ldots,\text{card } X\}$ such that $B_{q+1} = R \bullet B_q = B_q$, further $B_q(x) \leq A(x)$ for all $x\in X$ and $A\in F(X)$ satisfying Equation (4), that is $B_q$ is the SEFS of R. The following illustrative example makes clear the above concepts.

**Example:** Let cardX = 8 and $R\in F(X\times X)$ be the following fuzzy relation:

$$R = \begin{pmatrix} 0.6 & 0.7 & 0.4 & 0.3 & 0.5 & 0.6 & 0.3 & 0.6 \\ 0.8 & 1.0 & 0.3 & 0.5 & 0.2 & 0.8 & 0.0 & 0.3 \\ 0.7 & 0.8 & 0.2 & 0.4 & 0.6 & 0.4 & 0.4 & 0.2 \\ 1.0 & 0.5 & 0.2 & 0.3 & 0.6 & 1.0 & 0.1 & 0.1 \\ 0.4 & 0.9 & 0.4 & 0.5 & 0.1 & 0.9 & 0.2 & 0.5 \\ 0.9 & 0.6 & 0.2 & 0.2 & 0.4 & 0.7 & 0.3 & 0.4 \\ 0.5 & 0.5 & 0.2 & 0.3 & 0.3 & 0.5 & 0.2 & 0.3 \\ 0.4 & 0.6 & 0.3 & 0.2 & 0.1 & 0.2 & 0.1 & 0.1 \end{pmatrix}.$$

By using the sequence (3), we have that $A_2$ is the GEFS of R since

$$A_1 = (1.0, 1.0, 0.4, 0.5, 0.6, 1.0, 0.4, 0.6),$$

$$A_2 = (0.8, 1.0, 0.4, 0.5, 0.5, 0.8, 0.4, 0.6),$$

$$A_3 = R \text{ o } A_2 = A_2.$$

By using the sequence (6), we have that $B_1$ is the SEFS of R since

$$B_1 = (0.4,0.5,0.2,0.2,0.1,0.2,0.1,0.1), \ B_2 = R \bullet B_1 = B_1.$$

# 3 The Genetic Algorithm and its Application to Image Reconstruction

A genetic algorithm (for short, GA) is generally used to approximate global solutions of an optimization problem. The genetic algorithms are a particular class of evolutionary algorithms that simulate techniques based on biological evolution theory such as inheritance, mutation, selection and crossover. In a GA we consider a population of candidate solutions called individuals or phenotypes, that evolve toward the best solution of an optimization problem. The basic chromosome representing an individual is formed from a binary string, however it is possible to use other encodings. The initial population of chromosomes is generated randomly and a fitness value is calculated in each generation for each individual of the population. By basing on this fitness value, multiple individuals are stochastically selected from the current population and modified with the crossover and mutation operators and thus they form a new population, which in turn is used in the next iteration of the algorithm. Generally speaking, the GA terminates when either a maximum number of generations has been produced or a satisfactory fitness level has been reached for the population. In a GA the main step is the definition of a solution encoding, that is a chromosome must contain information about the solution that it represents. The encoding (binary strings, integers, real numbers, parsing trees, etc.) depends mainly on the problem under study.

In our method the population of chromosomes is formed by random images and the gene of a chromosome is a pixel, which assumes a value belonging to the set {0,1,…,L} (in our tests we assume L = 255). The comparison of the original image R with the k-th image-chromosome $R_k$ is made with the following distance:

$$d(R,R_k) = \sum_{x\in X} \left((A(x)-A_k(x))^2 + (B(x)-B_k(x))^2\right) \tag{7}$$

where X = {1,2,…,N}, k∈{1,…, M}, being M the cardinality of the initial population of the image-chromosomes, A and B (resp. $A_k$ and $B_k$) are the GEFS and SEFS of the fuzzy relation R (resp. $R_k$) obtained by normalizing the pixels of the original image (resp. the k-th image-chromosome $R_k$ of the population). We applied the distance (7) to derive the following fitness function F: $R_k \rightarrow [0,\infty)$, k∈{1,…, M}, defined as

$$F(R_k) = \frac{1}{d(R,R_k)}. \tag{8}$$

In the population of the mating pool we have used the roulette-wheel selection method: the probability $P_S(R_k)$ of the k-th image-chromosome $R_k$ is given by

$$P_s(R_k) = \frac{F(R_k)}{\sum_{j=1}^{M} F(R_j)}. \tag{9}$$

Afterwards the crossover operator is applied to the population of the mating pool for the generation of a child from two parent chromosomes with probability $p_{cross}$. Figure 1 shows how we apply the single point fixed-lenght bidimensional array crossover operator to two image-chromosomes in the mating pool.

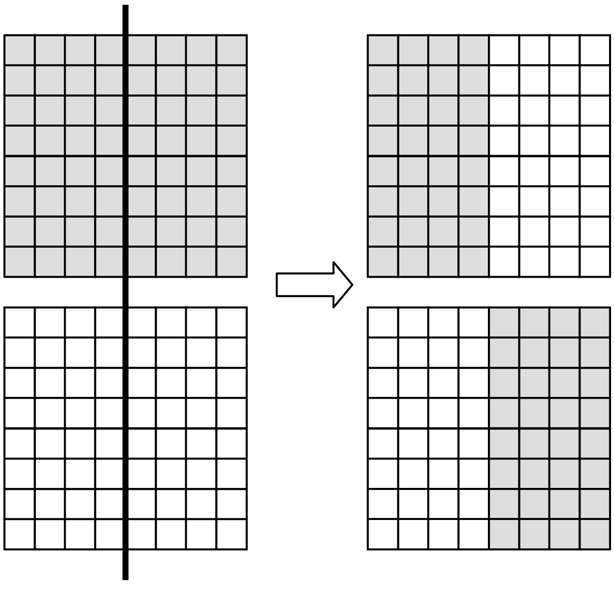

Figure 1: The single point fixed-length bidimensional array crossover operator

Successively we apply a mutation flip operator to each chromosome with probability p $_{\text{mut}}$, such operator performs the random change of two genes in a chromosome. During the selection process, we choose the chromosomes with the highest fitness value (8) in the mating pool. The fitness, selection, crossover and mutation operators are iterated until the number “ Iter” of iterations is equal to a given maximum value “I(Max)”. The procedure is presented in Figure 2 and is given in the following pseudocode:

```
Calculate GEFS and SEFS of the fuzzy relation R (input image)
Create the initial population of chromosomes $R_k$, k=1,…, M
While Iter = Number of Iterations ≤ I(Max) = Max Number of Iterations
    For each $R_k$ in the population:
        - Calculate GEFS and SEFS of the fuzzy relation $R_k$
        - Calculate the fitness value $F(R_k)$
        - Calculate the selection probability value $P_S(R_k)$
    Next
    - Populate the mating pool
    - In the mating pool apply the bidimensional array crossover operator with
      probability $p_{cross}$ to each couple $(R_k,R_h)$
    - In the mating pool apply the mutation flip operator with probability $p_{mut}$ to each $R_k$
End while
```

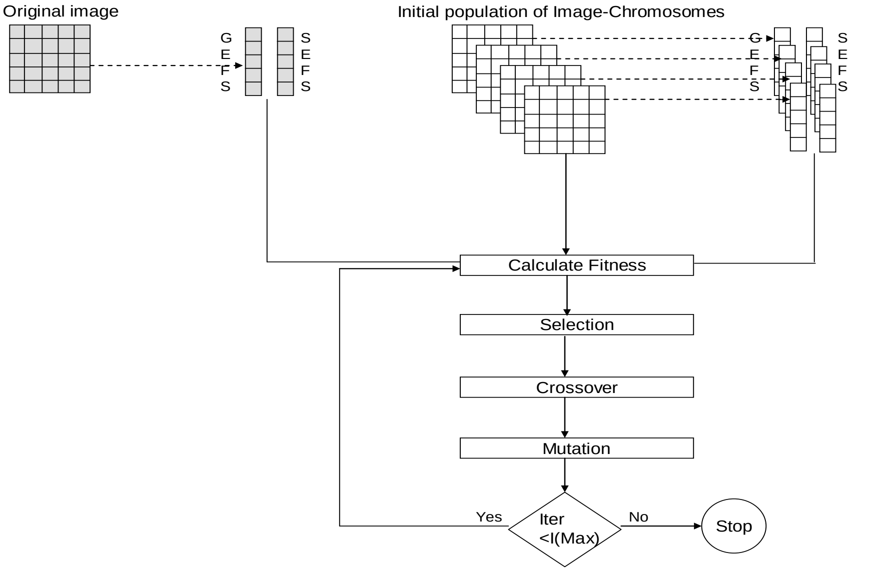

Figure 2: Design of the image reconstruction process

Now we consider the computational complexity of our GA: if the image to be reconstructed has sizes N × N and we are faced with a initial population of M image-chromosomes, we have a computational complexity equal to O(Iter×M×N×N) after a number Iter (≤ I(Max)) of iterations.

## 4 A Convex Combination of Max-min and Min-max Operators

For sake of completeness, here we describe the main tool of [8]. Indeed, let X be a finite referential set, $R\in F(X \times X)$, $A\in F(X)$ and $\lambda\in[0,1]$ such that

$$\lambda\cdot(R \text{ o } A) + (1 - \lambda)\cdot(R \bullet A) = A, \tag{10}$$

that is we consider a convex combination of max-min and min-max operators and A is said to be an eigen fuzzy set of R with respect to such convex combination. Note that if $\lambda = 0$ (resp. $\lambda = 1$), Equation (10) becomes Equation (1) (resp. Equation (4)). For R given, we assume A as unknown in Equation (10). Then Theorem 4 of [8] assures that there exists the smallest eigen fuzzy set A with respect to the above convex combination defined pointwise as

$$A(x) = \lim_{n\to\infty} A_n(x) \tag{11}$$

for all x∈X, where the sequence of fuzzy sets $\{A_n : n = 0,1,2\ldots\}$ is defined from the following recursive formula:

$$A_0(x) = 0 \text{ for all } x\in X,\ A_{n+1} = \lambda\cdot(R \text{ o } A_n) + (1 - \lambda)\cdot(R \bullet A_n) \quad \text{for } n = 0,1,2,\ldots. \tag{12}$$

Dually one can obtain the greatest eigen fuzzy set with respect to the above convex combination. Let R be a fuzzy relation representing an original image and suppose that A is a known eigen fuzzy set of R with respect to the convex combination (10). The image reconstruction consists theoretically in the estimation of a fuzzy relation S∈F(X×X) satisfying the Equation (10). Unfortunately Theorem 4 of [8] is not constructive, that is it guarantees only the existence of such eigen fuzzy set A and does not give a way how to find A starting from R. Then the authors of [8] assume that R satisfies the following system of equations

$$A_n = \lambda_n\cdot(R \text{ o } A_n) + (1 - \lambda_n)\cdot(R \bullet A_n) \quad \text{for } n = 0,1,2\ldots,t \tag{13}$$

where $\lambda_n\in[0,1]$ is assigned and $A_n$ is defined from formula (12) for n = 0,1,2…,t. If S∈F(X × X) is an initial random fuzzy relation, then one can see to minimize the following cost function

$$Q(S)=\sum_{n=1}^{t}\sum_{x\in X}\left\{A_n(x)-\left[\lambda_n\cdot(S\text{o}A_n)(x)+(1-\lambda_n)\cdot(S\bullet A_n)(x)\right]\right\}^2 \tag{14}$$

by means of the gradient method. Indeed the following iterative formulas are useful:

$$S^{(1)}=S, S^{(i+1)}(x',y)=S^{(i)}(x',y)-\alpha\cdot\frac{\partial Q^{(i)}}{\partial S^{(i)}(x',y)}, i=1,2,\ldots \tag{15}$$

where $x'$, y∈X, $Q^{(i)}=Q(S^{(i)})$, “i” is the parameter which denotes the number of iterations, α is a learning rate and

$$\frac{\partial Q^{(i)}}{\partial S^{(i)}(x',y)}=2\cdot\sum_{i=1}^{t}\sum_{x\in X}\left\{A_n(x)-\left[\lambda_n\cdot\left(S^{(i)}\circ A_n\right)(x)+(1-\lambda_n)\cdot\left(S^{(i)}\bullet A_n\right)(x)\right]\right\} \\ \cdot\left\{-\lambda_n\cdot\frac{\partial}{\partial S^{(i)}(x',y)}\left(S^{(i)}\circ A_n\right)(x)-(1-\lambda_n)\cdot\frac{\partial}{\partial S^{(i)}(x',y)}\cdot\left(S^{(i)}\bullet A_n\right)(x)\right\}. \tag{16}$$

Furthermore the first partial derivative at second member of (16) is given from

$$\frac{\partial}{\partial S^{(i)}(x',y)}\left(S^{(i)}\circ A_n\right)(x)=\frac{\partial}{\partial S^{(i)}(x',y)}\max_{x\in X}\left\{\min\left\{A_n(x),S^{(i)}(x,y)\right\}\right\}= \\ \phi\left(\max_{x\in X-\{x'\}}\left\{\min\left\{A_n(x),S^{(i)}(x,y)\right\}\right\},\min\left\{A_n(x'),S^{(i)}(x',y)\right\}\right)\cdot\psi\left(A_n(x'),S^{(i)}(x',y)\right) \tag{17}$$

where the functions φ and ψ are defined, for all a,b∈[0,1], as φ(a,b)=1 if a ≥ b, φ(a,b)= 0 if a < b and ψ(a,b)=1 if a ≤ b, ψ(a,b)=0 if a > b. Dually one can establish a similar formula for the second partial derivative at second member of (16). Of course the process is performed until the inequality $|Q^{(i+1)} - Q^{(i)}| < \varepsilon$ is satisfied for some index j, ε being an assigned threshold. The best results are obtained by setting t=16 or t=8 for a monochromatic image of sizes 256×256. In [8] the authors define also another method based on permutation fuzzy matrices, which is less efficient with respect to the above method and hence here not presented.

## 5 Results of the Experiments

We have used 350 images of sizes 256× 256 extracted from the well known SIDBA image dataset (http://www.cs.cmu.edu/~cil/vision.html) consisting of 10000 images. For our experiments we have implemented a C++ project with the C++ library class GALIB (http://lancet.mit.edu/ga/dist/). For brevity, we show our results only for the two images “Lena” and “Bird”. In our experiments we put $p_{cross} = 0.6$ and $p_{mut} = 0.01$; furthermore we utilize several values of M and I(Max). For the reconstructed image S (= $R_k$ for some k∈{1,2,…,M}), we give the classical Peak Signal to Noise Ratio (PSNR), calculated with respect to the original image R, as

$$PSNR(R,S)=20\log_{10}\left(\frac{255}{\sqrt{MSE(R,S)}}\right) \tag{18}$$

where MSE(R,S) is the Mean Square Error defined as

$$MSE(R,S)=\frac{\sum_{x\in X}\sum_{y\in X}\left[R(x,y)-S(x,y)\right]^2}{256\times256} \tag{19}$$

being x,y∈X={1,2,…,256}. In Table 1 we show the MSE(R,S) and PSNR(R,S) obtained for the gray image “Lena” with values of M=10, 40, 100 image-chromosomes and I(Max)=$10^3$, $5\times10^3$, $10^4$, $5\times10^4$, $10^5$.

Table 1: The values of MSE(R,S) and PSNR(R,S) obtained for “Lena”

| | M=10 | | M=40 | | M=100 | |
|---|---|---|---|---|---|---|
| I(Max) | MSE(R,S) | PSNR(R,S) | MSE(R,S) | PSNR(R,S) | MSE(R,S) | PSNR(R,S) |
| $10^3$ | 6808.95 | 9.80 | 6347.44 | 10.10 | 5878.32 | 10.43 |
| $5\times10^3$ | 6046.78 | 10.32 | 5411.31 | 10.80 | 3287.01 | 12.96 |
| $10^4$ | 5401.91 | 10.81 | 4909.99 | 11.22 | 1521.46 | 16.31 |
| $5\times10^4$ | 3788.76 | 12.34 | 3006.63 | 13.35 | 1098.43 | 17.72 |
| $10^5$ | 2347.88 | 14.42 | 1675.90 | 15.89 | 751.76 | 19.37 |

Figure 3a is the original image “Lena”. The Figures 3b, 3c, 3d, 3e show the recostructed images S for the respective values of M =10 and I (Max)=$10^3$, M= 40 and I (Max)=$10^4$, M =40 and I (Max)=$5\times10^4$, M =100 and I(Max)=$10^5$.

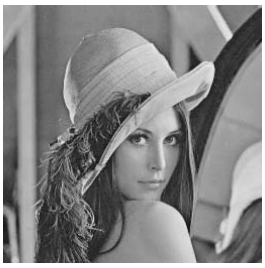

Figure 3a: Lena

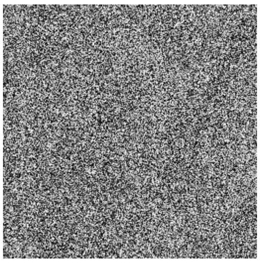

Figure 3b: M=10, I(Max)=103

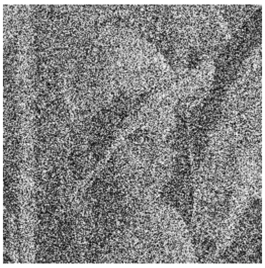

Figure 3c: M=40, I(Max)=104

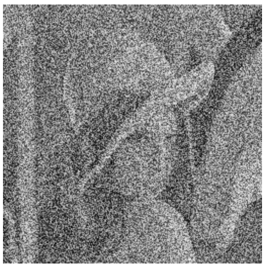

Figure 3d: M=40, I(Max)=5×104

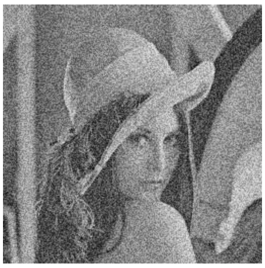

Figure 3e: M=100, I(Max)=105

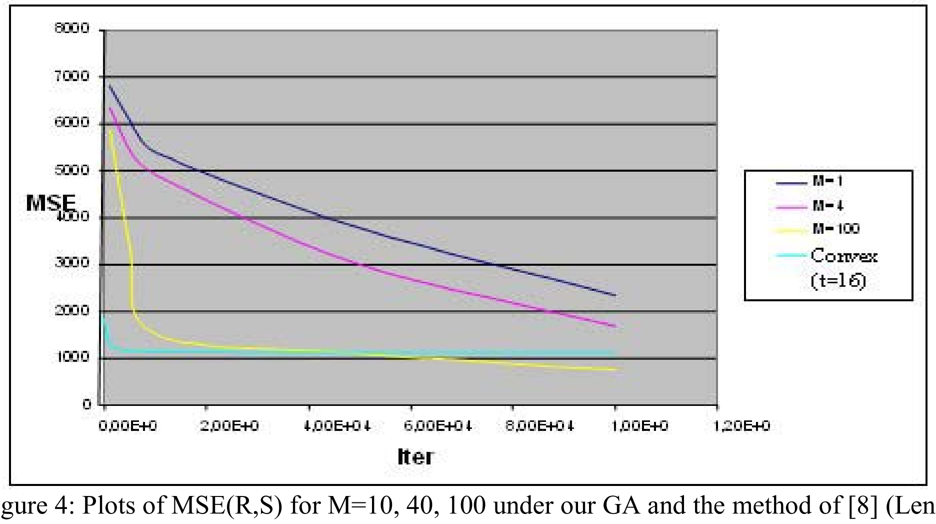

Figure 4: Plots of MSE(R,S) for M=10, 40, 100 under our GA and the method of [8] (Lena)

For several values of M, Figure 4 shows the trend of the MSE(R,S) obtained with respect to the iteration number Iter. For comparison, the MSE(R,S) obtained with the method of Section 4 is also plotted, in which we have assumed the same parameters of [8], that is $\lambda_n = n/(t-1)$ for n=0,1,…,t-1, t=16, $\alpha = 0.005$ and $\varepsilon = 10^{-8}$. By using an iteration number Iter $\geq 6\times10^4$ and M=100, in our GA we have an MSE(R,S) smaller than $10^3$ while the method of Section 4 gives an MSE(R,S) greater than $10^3$. Figure 5 (resp. Figure 6) contains the graphs of the GEFS (resp. SEFS) obtained for the original R of Figure 3a and for the reconstructed image S of Figure 3e. The distance is calculated with (7) and we have that d(R,S) = 0.987.

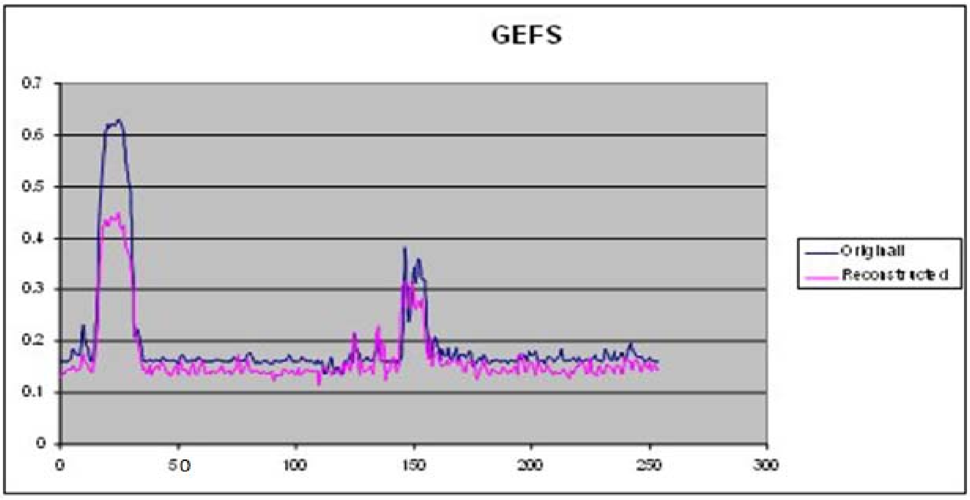

Figure 5: Plots of GEFS A for “Lena” (cfr. Figure 3a) and GEFS $A_k$ for the recon-structed image $S=R_k$ under M=100 and I(Max)=$10^5$ (cfr. Figure 3e)

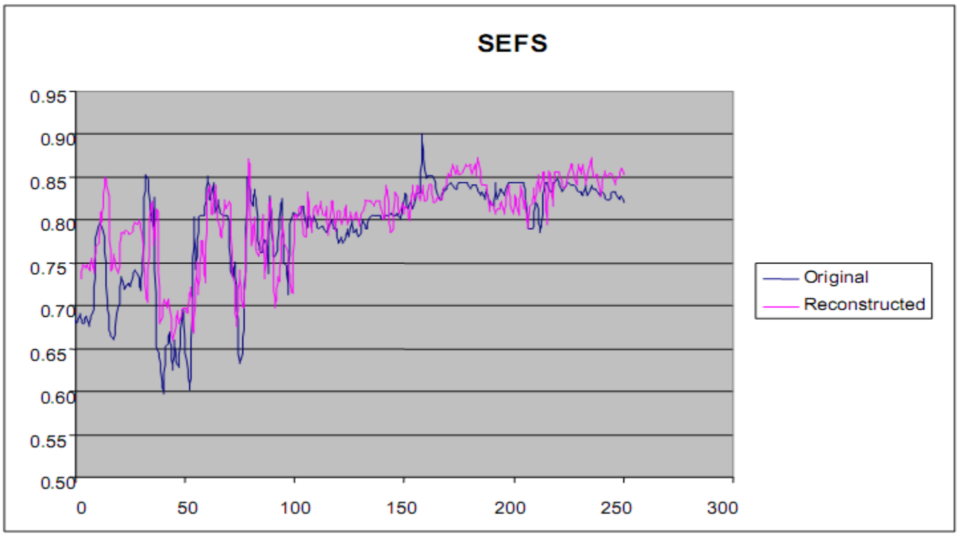

Figure 6: Plots of SEFS B for “Lena” (cfr. Figure 3a) and SEFS $B_k$ for the reconstructed image $S=R_k$ under M=100 and I(Max)=105 (cfr. Figure 3e)

In Table 2 we show the MSE(R,S) and PSNR(R,S) obtained for the gray image “Bird” with values of M=10, 40, 100 image-chromosomes and I(Max)=$10^3$, $5\times10^3$, $10^4$, $5\times10^3$, $10^5$. In corrispondence of the value M of the initial population, Figure 8 shows the trend of the MSE(R,S) obtained with respect to the iteration number I(Max).

Figure 7a is the original image “Bird”. The Figures 7b, 7c, 7d, 7e show the recostructed images S for the respective values of M=10 and I(Max)=$10^3$, M=40 and I(Max)=$10^4$, M=40 and I(Max)=$5\times10^4$, M=100 and I(Max)=$10^5$. Under the same parameters used for designing Figure 4, Figure 8 shows that the MSE(R,S) in our method is always smaller than that one calculated with the method of Section 4, in particular we have that MSE(R,S) $\leq 500$ if I(Max)$\geq10^5$. Figure 9 (resp. Figure 10) gives the related plots of GEFS (resp. SEFS) of the original image R of Figure 7a and of the reconstructed image S of Figure 7e. The distance is calculated with (7) and we have that d(R,S) = 0.570.

Table 2**:** The values of MSE(R,S) and PSNR(R,S) obtained for "Bird"

| | M=10 | | M=40 | | M=100 | |
|---|---|---|---|---|---|---|
| I(Max) | MSE(R,S) | PSNR(R,S) | MSE(R,S) | PSNR(R,S) | MSE(R,S) | PSNR(R,S) |
| $10^3$ | 6999.71 | 9.68 | 6454.91 | 10.03 | 5992.36 | 10.35 |
| $5\times10^3$ | 6189.77 | 10.21 | 5781.24 | 10.51 | 3678.47 | 12.47 |
| $10^4$ | 5744.85 | 10.54 | 5249.05 | 10.93 | 1521.46 | 16.31 |
| $5\times10^4$ | 3468.44 | 12.73 | 2648.12 | 13.90 | 879.14 | 18.69 |
| $10^5$ | 2169.70 | 14.77 | 1291.95 | 17.02 | 469.98 | 21.41 |

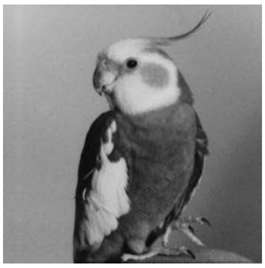

Figure 7a: Bird

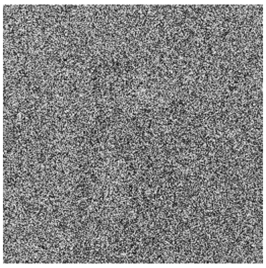

Figure 7b: M=10, I(Max) $=10^3$

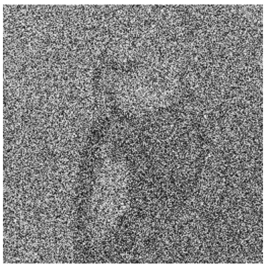

Figure 7c: M=40, I(Max)$=10^4$

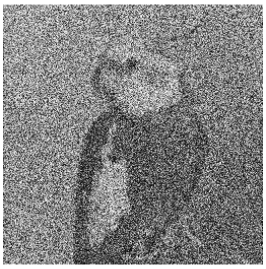

Figure 7d: M=40, I(Max)$=5\times10^4$

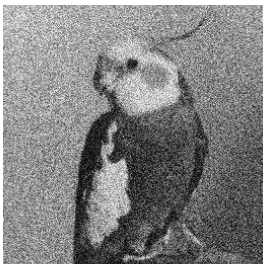

Figure 7e: M=100, I(Max)$=10^5$

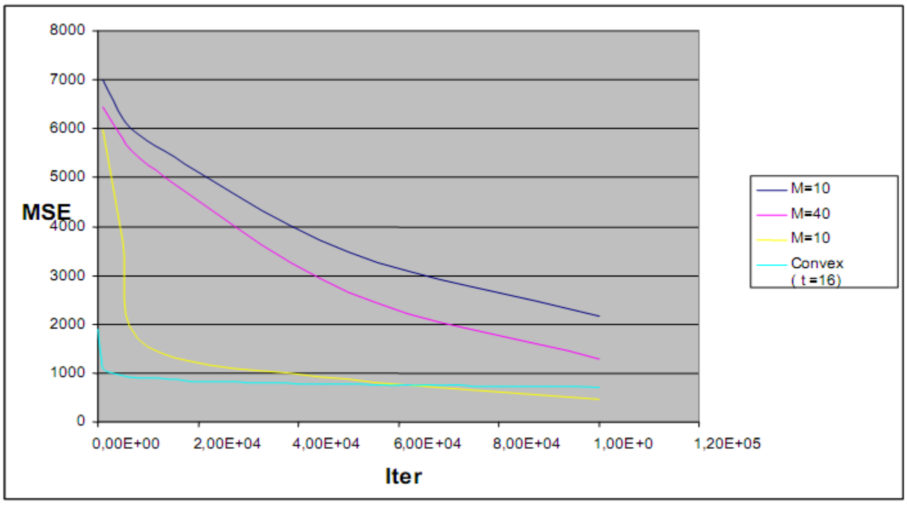

Figure 8: Plots of MSE(R,S) for M=10, 40, 100 under our GA and the method of [8] (Bird)

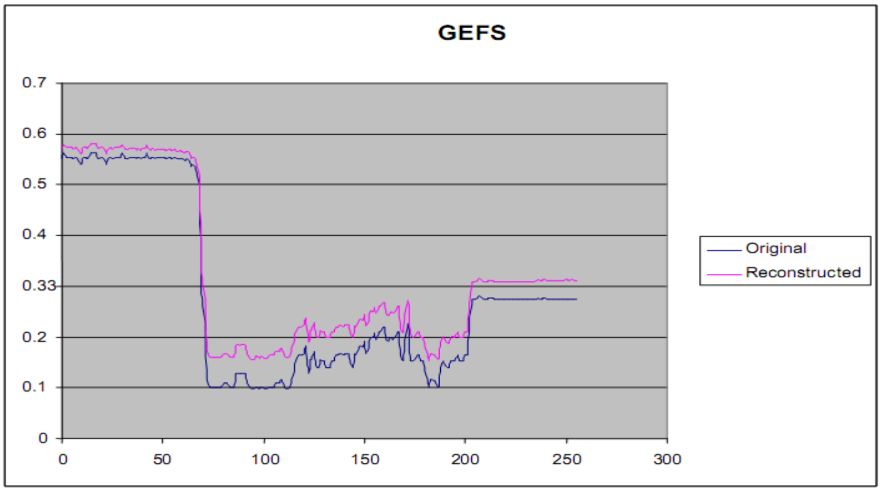

Figure 9: Plots of GEFS A for “Bird” (cfr. Figure 7a) and of GEFS $A_k$ for the reconstructed image $S=R_k$ under M=100 and I(Max)=$10^5$ (cfr. Figure 7e)

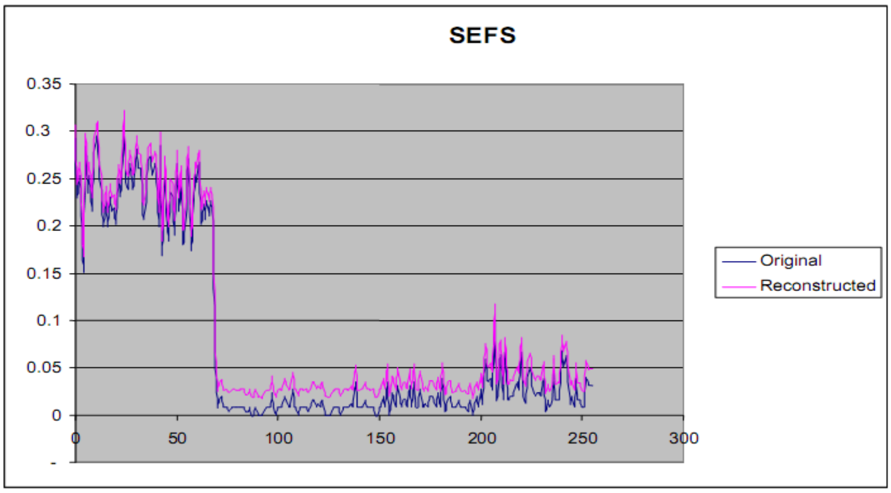

Figure 10: Plots of SEFS A for “Bird” (cfr. Figure 7a) and of SEFS $A_k$ for the reconstructed image $S=R_k$ under M=100 and I(Max)=$10^5$ (cfr. Figure 7e)

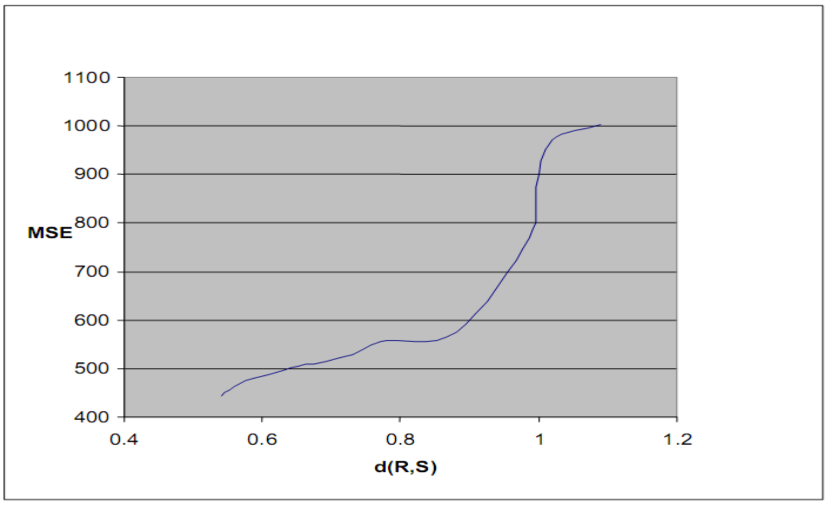

Figure 11: Plot of MSE(R,S) with respect to the distance d(R,S) for the 350 reconstructed images S by assuming M=100 and I(Max) =$10^5$

By using a sample of 350 images of sizes 256×256 extracted from SIDBA image dataset, we have plotted in Figure 11 the value of the MSE(R,S) with respect to the distance d (R,S) under the assumptions M =100 and I(Max)=$10^5$. We observe that MSE(R,S) increases if the distance d(R,S) increases by resulting MSE(R,S) $\leq 10^3$. Therefore we obtain better results than those ones presented in [8] by using in our GA a initial population of 100 image-chromosomes and $10^5$ iterations, with a computational complexity equal to O($10^5 \times 100 \times 256 \times 256$).

## 6 Conclusions

We have experimented a GA over 350 gray images of sizes 256 × 256 extracted from the well known SIDBA image dataset. Two types of eigen fuzzy sets (GEFS and SEFS) with respect to the max-min and the min-max operators are used for the calculation of the fitness value in the GA. We have used several values of the initial population of image-chromosoms and we have assumed as reconstructed image that image-chromosome with the greatest fitness value. In other words, we take the image-chromosome with its GEFS and SEFS very close to those ones of the original image. Our method gives better results with respect to the other method [8] based on a combination convex of both GEFS and SEFS. Other applications of the concepts here presented shall be made in future papers.

## References

[1] Bourke, M.M., and D. Grant Fisher, Convergence, eigen fuzzy sets and stability analysis of relation matrices, *Fuzzy Sets and Systems*, vol.8, pp.227–234, 1996.

[2] Buckley, J.J., Fuzzy eigenvalues and input-output analysis, *Fuzzy Sets and Systems*, vol.34, pp.187–195, 1990.

[3] Di Martino, F., H. Nobuhara, and S. Sessa, Eigen fuzzy sets and image information retrieval, *Handbook of Granular Computing*, edited by W. Pedrycz, A. Skowron and V. Kreinovich, John Wiley &Sons, Chichester, England, 2008.

[4] Di Martino, F., and S. Sessa, Digital watermarking in coding/decoding processes with fuzzy relation equations, *Soft Computing*, vol.10, pp.233–243, 2006.

[5] Jacas, J., and J. Recasens, Fuzzy T-transitive relations: Eigenvectors and generators, *Fuzzy Sets and Systems*, vol.72, pp.147–154, 1995.

[6] Loia, V., W. Pedrycz, and S. Sessa, Fuzzy relation calculus in the compression and decompression of fuzzy relations, *International Journal of Image and Graphics*, vol.2, pp.1–15, 2002.

[7] Loia, V., and S. Sessa, Fuzzy relation equations for coding/decoding processes of images and videos, *Information Sciences*, vol.171, pp.145–172, 2005.

[8] Nobuhara, H., B. Bede, and K. Hirota, On various eigen fuzzy sets and their application to image reconstruction, *Information Sciences*, vol.176, pp.2988–3010, 2006.

[9] Nobuhara H., and K. Hirota, A solution for eigen fuzzy sets of adjoint max-min decomposition and its application to image analysis, *Proceeding of WISP, IEEE Internat. Symp. on Intelligent Signal Processing*, IEEE, Budapest, pp.27–30, 2003.

[10] Nobuhara, H., K. Hirota, and W. Pedrycz, Fast solving method of fuzzy relational equations and its application to lossy image compression, *IEEE Transactions of Fuzzy Systems*, vol.8, no.3, pp.325–334, 2000.

[11] Nobuhara, H., K. Hirota, and W. Pedrycz, Relational image compression: optimizations through the design of fuzzy coders and YUV color space, *Soft Computing*, vol.9, no.6, pp. 471–479, 2005.

[12] Nobuhara, H., K. Hirota, W. Pedrycz, and S. Sessa, Two iterative methods of decomposition of a fuzzy relation for image compression/decompression processing, *Soft Computing*, vol.8, no.10, pp. 698–704, 2004.

[13] Pedrycz, W., K. Hirota, and S. Sessa, A decomposition of fuzzy relations, *IEEE Transaction on Systems, Man, and Cybernetics*, vol.31, no.4, pp.657–663, 2001.

[14] Pedrycz, W., and M. Reformat, Genetic optimization with fuzzy coding, *Genetic Algorithms and Soft Computing*, edited by F. Herrera and J. Verdegay, Physica Verlag, Berlin, pp.51–67, 1996.

[15] Sanchez, E., Resolution of eigen fuzzy sets equations, *Fuzzy Sets and Systems*, vol.1, pp.69–74, 1978.

[16] Sanchez, E., Eigen fuzzy sets and fuzzy relations, *Journal of Mathematical Analysis and Applications*, vol.81, pp.399–421, 1981.

[17] Wagenknecht, M., and K. Hartmann, On the construction of fuzzy eigen solutions in given regions, *Fuzzy Sets and Systems*, vol.20, pp.55–65, 1986.

[18] Wagenknecht, M., and K. Hartmann, Eigen fuzzy number sets, *Fuzzy Sets and Systems*, vol.16, pp.75–85, 1985.