Nature inspired metaheuristics for improved JPEG steganalysis

Anita Christaline. J[1] **· Ramesh. R**[2] **· Gomathy. C**[1] **·**
Vaishali. D[1]

Received: 2 February 2017 / Revised: 14 June 2017 / Accepted: 22 June 2017 /
Published online: 10 July 2017

Abstract The performance accuracy of JPEG steganalysis depends on the best features extracted from the images. This demands extraction of all possible features that undergo changes during embedding. The computational complexity due to such large number of features necessitates feature set optimization. Existing research in JPEG image steganalysis tend to extract rich feature sets and reduce them by statistical feature reduction techniques. Compared to these techniques, genetic algorithm based optimization techniques are more promising as they converge to global minima. The objective of this paper is to implement genetic based optimization to reduce the high dimensional image features and hence obtain improved classification accuracy. The method implemented includes the extraction of image features in terms of co-occurrence matrices of the differences of all possible Discrete Cosine Transform (DCT) coefficients to give 200 × 23,230 features. These features are optimized by a nature inspired meta-heuristic, Ant Lion Optimization (ALO) which considers the features as ants that move in random space. The fitness function for the antlion to hunt the ants is proportional to the traps built by the antlion. The proposed steganalyser has been tested for classification accuracies with different payloads. The classifiers implemented include Support Vector Machines (SVM), Multi Layer Perceptron (MLP) and fusion classifiers based on Bayes, Decision template and Dempster Schafer data fusion schemes. The results show that

✉ Anita Christaline. J
anita.j@vdp.srmuniv.ac.in

Ramesh. R
ramesh@saveetha.ac.in

Gomathy. C
hod.ece@vdp.srmuniv.ac.in

Vaishali. D
vaishali.b@vdp.srmuniv.ac.in

[1] Department of ECE, SRM University, Chennai, India

[2] ECE Department, Saveetha Engineering College, Chennai, India

highest average classification accuracy has been obtained for Bayes fusion classifier followed by Dempster Schafer fusion classifier. It has been noted that the performance of fusion classifiers is better compared to individual classifiers. Thus the proposed method gives better classification accuracy for JPEG steganalysis than existing methods.

Keywords JPEG steganalysis · Fusion classifiers · Ant Lion Optimization · Bayes fusion classifier

1 Introduction

Image steganography is the mechanism of embedding secret information into a clean (cover) image using a stego key. The embedded image is called as the stego image. Owing to the fact that human eye is less perceptive to slight changes in image parameters, image steganography has gained importance and is widely used by terrorists. The common steganographic algorithms include JSteg, JP Hide and Seek, Outguess. Apart from these commercial tools, many application specific steganographic algorithms have been developed by researchers.

While steganography relates to secret communication, steganalysis involves hacking a suspicious image and identifying any secret message. There are two methods of image steganalysis, embedding specific and blind steganalysis. The embedding specific methods know the steganographic algorithm that was used to embed, but blind steganalysis does not have knowledge about the embedding logic that was used. Hence blind steganalysis is called as Universal method where the suspected image is classified by a two class (clean or stego) classifier. This classification could be based on statistical methods or computational intelligence methods [25]. While statistical blind steganalysis use Markov or wavelet or DCT coefficients to classify an image, computational intelligence methods are based on neural networks and genetic or evolutionary algorithms [9]. The efficiencies of these methods depend on the optimal features selected from the image for classification. Researchers state that acquiring the best feature set has always been a challenge as different steganographic algorithms change different image parameters [14, 20, 23, 29]. Recent research has presented a method of acquiring a rich set of image features that would give best classification of images but have disadvantages in terms of computational complexity [12]. This demands the need to optimize the data sets for best possible features. The most prominent statistical feature reduction methods convergence to local minima and choose inappropriate features.

To overcome this challenge this research intends to implement nature inspired optimization techniques for best image steganalysis. These are meta-heuristic algorithms that use stochastic operators to obtain global optimal values [2]. Randomness being the main characteristics of these stochastic algorithms, they search for the global optimum by creating a set of candidate solutions and then iteratively fine tune them till a satisfactory terminating condition [28]. The recent nature inspired algorithms include Firefly Algorithm (FA) [31], Cuckoo Search (CS) algorithm [32], Cuckoo Optimization Algorithm (COA) [24], Ray Optimization (RO) algorithm [15–17]. This research work implements modified Ant Lion Optimization for steganalysis of JPEG images. The next section discusses the proposed steganalyser, followed by the extracted image features and then ALO optimization technique. Finally the classification scheme used for steganalysis is discussed.

2 Implementation of proposed steganalyser

The proposed image steganalyser in this research consists of four main parts. In the first part stego images are created from clean images using the non shrinkage F5 (nsF5) algorithm. The second part involves the extraction of rich feature set (23,230 features) from stego and clean images. Optimizing this feature set to 232 features by ALO is the third part, followed by classification in forth stage. All implementations are done in MATLAB.

2.1 Image database and steganographic embedding

The images used in this research are taken from the standard BOSS (Break Our Steganographic System) database [1]. This database has full resolution images acquired from different cameras like Panasonic, Leica, Pentax, Cannon EOS and Nikon in .cr2 or .dng format and later converted into .pgm format. The original images are resized and cropped to 512 × 512. These images are made public for use by researchers in the field of steganography and steganalysis. From the images available in this standard database (http://dde.binghamton.edu/download/) 100 images are chosen and converted into .jpg format with a quality factor of 75. These 100 images are the cover (clean) images that are used to create another 100 stego images by Non Shrinkage F5 (nsF5) embedding. The subsequent feature extraction, optimization and classification in this research uses these 200 images.

The stego images for this research work have been created from the Non Shrinkage F5 (nsF5) algorithm. Permutation straddling of the image pixels in F5 enables uniform embedding of the data with a specific time complexity of order *O(n)*. The sequences of operations involved in F5 are – choose a DCT coefficient, permute it with a key, embed the secret data and then code it with Huffman coding. For s secret bits and code words with length $n = 2^s - 1$, the embedding rate *R(s)*, change in embedding density *E(s)* and embedding efficiency *(μ)* are

$$R(s) = \frac{s}{n} = \frac{ks}{2^s - 1} \tag{1}$$

$$E(s) = \frac{1}{n+1} = \frac{1}{2^s} \tag{2}$$

$$\mu = \frac{R(s)}{S(s)} = \frac{s2^s}{2^s - 1} \tag{3}$$

Coefficients that become zero due to embedding are avoided in F5 as the decoder in receiver skips that coefficient. Hence the modified F5 (non shrinkage F5) algorithm uses syndrome coding on the DCT coefficients before applying F5 logic. This method eliminates the shrinkage problem and is superior compared to other steganographic methods that use side information about the cover. Literature shows that nsF5 algorithm is the best steganographic algorithm till date for JPEG images [18].

2.2 Extracted image features

Features extracted in this research include a rich set of all possible changes in DCT coefficients, that have spatial and frequency correlations. For a JPEG image of dimension $I \times J$, the quantized DCT coefficients could be in a matrix $C \in X^{I \times J}$. Each DCT coefficient $C_{p,q}{}^{m,n}$ represents the $(p,q)^{th}$ coefficient in the $(m,n)^{th}$ 8 × 8 block where (p,q) ∈ $\{0,1,....7\}^2$, m ∈ {1,2,....I/8} and n ∈ {1,2,.... J/8}. For simplicity the individual elements are represented as $C_{m,n}$. The matrices denoting the absolute values, inter block differences and intra block differences are

$$DC_{m,n} = |C_{m,n}|, m = 1,2,\ldots I \, and \, n = 1,2,\ldots J \tag{4}$$

$$DC^{ha}{}_{m,n} = |C_{m,n}| - |C_{m,n+1}|, m = 1,2,\ldots I \, and \, n = 1,2,\ldots J-1 \tag{5}$$

$$DC^{va}{}_{m,n} = |C_{m,n}| - |C_{m+1,n}|, m = 1,2,\ldots I-1 \, and \, n = 1,2,\ldots J \tag{6}$$

$$DC^{da}{}_{m,n} = |C_{m,n}| - |C_{m+1,n+1}|, m = 1,2,\ldots I-1 \, and \, n = 1,2,\ldots J-1 \tag{7}$$

$$DC^{he}{}_{m,n} = |C_{m,n}| - |C_{m,n+8}|, m = 1,2,\ldots I \, and \, n = 1,2,\ldots J-8 \tag{8}$$

$$DC^{ve}{}_{m,n} = |C_{m,n}| - |C_{m+8,n}|, m = 1,2,\ldots I-8 \, and \, n = 1,2,\ldots J \tag{9}$$

where, $DC_{m,n}$ represents difference of absolute value, $DC^{ha}{}_{m,n}$ is the difference between the intra block horizontal coefficients, $DC^{va}{}_{m,n}$ is the difference between the intra block vertical coefficients, $DC^{da}{}_{m,n}$ is the difference between the intra block diagonal coefficients, $DC^{he}{}_{m,n}$ is the difference between the inter block horizontal coefficients and $DC^{ve}{}_{m,n}$ is the difference between the inter block vertical coefficients. The proposed feature model is the sub model framed from the 2D co-occurrence matrices calculated from these difference coefficients. There are 10 sub models in each of the co-occurrences matrices and final rich feature set has 23,230 features from all possible co-occurrence combinations of DCT coefficients. The details of the features due to each of the sub bands are enumerated in Table 1.

This rich feature set has 23,230 features from all possible co-occurrence combinations of DCT coefficients.

Table 1 The Extracted feature Model

Feature model (co-occurrences)	No. of features
Absolute values of DCT coefficients	2512
Differences in absolute values - horizontal and vertical - intra block coefficients	2041
Differences in absolute values – diagonal - intra block coefficients	2041
Differences in absolute values - horizontal and vertical - inter block coefficients	2041
Integral co-occurrences from absolute values	900
Integral co-occurrences from differences of absolute values	2440
DCT mode specific absolute components of reference	2512
Co-occurrences of differences of absolute values in horizontal and vertical directions of reference	2041
Inter block horizontal and vertical absolute values of reference	2041
Co-occurrences of differences in absolute values in diagonal direction of reference	2041
Integral co-occurrences of differences in absolute values of reference	2620

2.3 ALO optimization

With such a large and diverse feature set, classification is a time and space complex problem. Hence the feature set is reduced by Ant Lion Optimization, a nature inspired meta-heuristic algorithm. The positions of the ants are the features in the search space in a specific iteration. The characteristics of the ALO algorithm are

- The movements of the ants and antlions are random walks in search space.
- The fitness function depends on the size of the pits built by the antlions (probability of ants getting trapped is more if the pit is larger in size).
- In each iteration an elite antlion catches a prey.
- Fitter ants are caught by the antlion (fitter features are selected).
- Antlions start building new pits by repositioning themselves to the latest prey.

Mathematical representation of the random walk of ants is

$$R(t) = [0, CU(2s(t_1)-1, CU(2s(t_2)-1, CU(2s(t_3)-1, \ldots\ldots CU(2s(t_n)-1] \qquad (10)$$

Here *'CU'* is the cumulative sum of random function *'s'* till *'n'* iterations, *'t'* is the step size of the random walk and *s(t)* is random function defined as 1 if $rand > 0.5$ and is 0 if $rand < 0.5$, where *rand* is a random number in the interval [0,1]. The nature of random walk is shown in Fig. 1 for three sets of features extracted with payload 0.5. The nature of randomness can be seen from the fluctuation in the curves around the origin (as expected for the behaviour of ants in search space).

The position of each ant is stored in matrix format for use in subsequent iterations during optimization.

$$\text{AntPos} = \begin{bmatrix} A1,1 & A1,2\ldots\ldots A1,p \\ A2,1 & A2,2\ldots\ldots A2,p \\ & \vdots \\ An,1 & An,2\ldots\ldots An,p \end{bmatrix} \qquad (11)$$

here, $A_{n,p}$ is the p^{th} position of the n^{th} ant in any iteration. The number of features is *n,* corresponding to ants and their *p* positions correspond to *p* features. Each ant

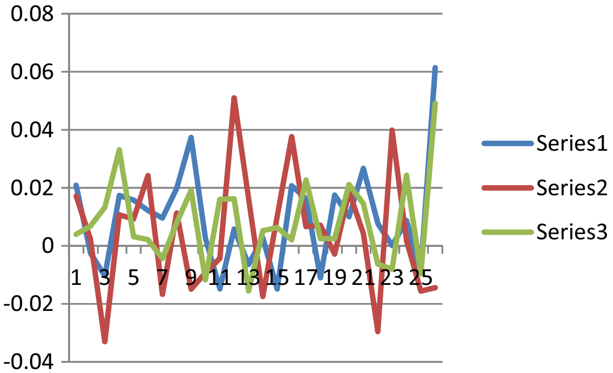


Fig. 1 Three set of feature values depicting the random walk of three ants in search space

(feature vector) is optimized according to a fitness function and the fitness values are stored in a matrix,

$$\mathrm{AntOpt} = \begin{bmatrix} Aopt1 \\ Aopt2 \\ \vdots \\ Aoptn \end{bmatrix} = \begin{bmatrix} f(\mathrm{A1,1} & \mathrm{A1,2} & \mathrm{A1,p}) \\ f(\mathrm{A2,1} & \mathrm{A2,2} & \mathrm{A2,p}) \\ & \vdots & \\ f(\mathrm{An,1} & \mathrm{An,2} & \mathrm{An,p}) \end{bmatrix} \tag{12}$$

here, $f(An, 1\ An, 2\ An, p)$ is the function for calculating the optimal value of the n$^{\mathrm{th}}$ ant. According to the ALO algorithm, apart from the ants, antlions also hide in the search space to catch the ants. Their positions need to be tracked and are stored in another matrix,

$$\mathrm{AntLionPos} = \begin{bmatrix} L1,1 & L1,2\ldots\ldots L1,d \\ L2,1 & L2,2\ldots\ldots L2,d \\ & \vdots \\ Ln,1 & Ln,2\ldots\ldots Ln,d \end{bmatrix} \tag{13}$$

where, $L_{n,p}$ is the d^{th} position of the n^{th} antlion. The n antlions have d positions for d features. The fitness values of antlions are stored in a matrix,

$$\mathrm{AntLionOpt} = \begin{bmatrix} Lopt1 \\ Lopt2 \\ \vdots \\ Loptn \end{bmatrix} = \begin{bmatrix} f(\mathrm{L1,1} & \mathrm{L1,2} & \mathrm{L1,d}) \\ f(\mathrm{L2,1} & \mathrm{L2,2} & \mathrm{L2,d}) \\ & \vdots & \\ f(\mathrm{Ln,1} & \mathrm{Ln,2} & \mathrm{Ln,d}) \end{bmatrix} \tag{14}$$

here, $f(Ln, 1\ Ln, 2\ Ln, d)$ is the function for calculating the optimal value of the n$^{\mathrm{th}}$ antlion. The phenomenon of random walk may be diverse, hence to accommodate within the search space, the variables are normalized by min-max normalization technique as below.

$$A = \frac{(A-Mn) \times (B-Ct)}{(Bt-Mn)} + c \tag{15}$$

Mn are *B* are the minimum and maximum values of n^{th} feature vector, *Ct* is minimum of feature vector in n^{th} iteration and *Bt* is the maximum value of feature in n^{th} iteration. Normalization ensures that all the feature values are in the search space. During the random walk, the ants get trapped in an antlion's pit. Figure 2 shows the pits built by one or more antlions.

Fig. 2 Pits built by one or more Antlions

A pit is modelled as a hyper sphere for each selected antlion and the movement of ants around the antlion in the hyper sphere is modelled according to the following equations,

$$C_i^{\ t} = L_n^{\ t} + C^t \tag{16}$$

$$D_i^{\ t} = L_n^{\ t} + D^t \tag{17}$$

$C_i^{\ t}$ and $D_i^{\ t}$ are the minimum and maximum of the feature values of the i^{th} ant, C^t and D^t are the minimum and maximum among all feature values in the t^{th} iteration. $L_n^{\ t}$ is the n^{th} antlion in t^{th} iteration. When the ants come into the trap, they tend to move away, but the antlions spill or shoot sand upwards to make them slide. This sliding behaviour of the ants is modelled by decreasing radius of the hyper sphere defined by vectors C and D as

$$C^t = C^t/S \tag{18}$$

$$D^t \ = D^t/S \tag{19}$$

The parameter *S,* defines the level of accuracy [22] and is defined as

$$S = 10^{ut/T} \tag{20}$$

where t is the current iteration, T is the total number of iterations, u = 2 when t/T > 0.1, u = 3 when t/T > 0.5, u = 4 when t/T > 0.75, u = 5 when t/T > 0.9, u = 6 when t/T > 0.95. The values of C and D iteratively decrease mimicking the sliding of ants to the bottom of the pit (global optimal point). To acquire high probability of catching the next prey (ant), the antlions update their position to that of the hunted ant, if fitness of ant is greater than fitness of antlion.

In this research work the fitness function is sum of squares of feature values and the random walk of ants and antlions is implemented as roulette wheel selection. The number of search agents (antlions) is chosen as 100 and hence the rich image feature set of size 23,230 reduces to 232 after optimization. For 100 cover images and 100 stego images, the optimized feature set is 200 × 232. This reduced feature set is given to the fusion classifier system.

2.4 Fusion classifier system

The identification of an image as a clean or stego image is a two class problem (clean image is classified as 1 and a stego image is classified as 2). The classifiers used are individual classifiers (SVM and MLP) and their fusion by three schemes (Bayes, Decision template and Dempster Schafer). Fusion classifiers are superior as they exploit the strengths of individual classifiers but avoid their weaknesses [19]. The original feature set is divided into 3 folds. Two folds are used for training and 1 fold is used for validation.

Support Vector Machines fit the data into high dimensional feature space and separate them by hyperplane. SVM in this research uses RBF kernel with penalty factor 100 and gamma of 10. Multi Layer Perceptron (MLP) [30] is a feed forward network for non - linearly separable data. The MLP implemented in this research has 10 nodes to associate the inputs to high output response with sigmoidal activation function.

The classifier fusion methods implemented are Bayes, Decision template and Dempster Schafer [19]. These are decision based data fusion methods using the perceived data from many sources. Bayesian inference uses the conditional probability according to Bayes rule in terms of the posterior probability *P(Y/X)*, which demands the prior knowledge of *P(X)* and

Table 2 Embedding changes in stego images according to payload

Image file (Jpeg)	number of AC DCT coefficients in cover image	Payload (bpdct)	number of embedding changes
‘stego000’	91,487	0.01	78
		0.1	1190
		0.5	10,068
		0.8	22,233
‘stego073’	64,990	0.01	56
		0.1	846
		0.5	7152
		0.8	15,794

P(X/Y). Dempster Schafer fusion deals with uncertainty in terms of changing beliefs, evidences and incomplete knowledge. To combine the effect of two hypotheses (classifiers), the rule according to Dempster Schafer is in terms of mass functions or probabilities [19]. The selected classes try to maximize the belief function. Decision Template is another fusion scheme that combines classifiers by comparing the output of classifiers with a reference (decision) template. The reference templates are measured prior to comparison. The comparison is based on similarity measure and consistency measure. This method differs from other methods in that it considers the output of all classifiers to make the final support for a class, while other methods consider the output of that class alone to calculate the support [19]. This enables a decision based on the average of decision profiles due to all elements in the training set. Classification accuracy is used to compare the performance of individual and fusion classifiers. Accuracy is *(Tp + Tn)/ (Tp + Tn + Fp + Fn), Tp* is True Positive or hit, *Tn* is True Negative or rejection, *Fp* is false Positive or false alarm and *Fn* is False Negative or miss.

3 Results

The stego image is obtained for specific payloads on randomly selected DCT coefficients of cover images with a PRNG seed as in Table 2.

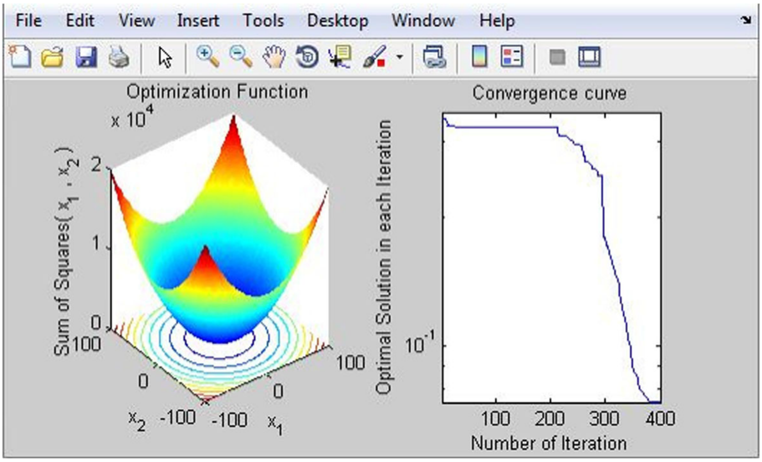


Fig. 3 Convergence of ALO Optimization

Table 3 Classification accuracy of single and fusion classifiers for Payload = 0.5

Original Feature set size	Optimized Feature set size	Payload during Embedding	PRNG seed during embedding	SVM	MLP	Bayes	Decision Template	Dempster Schafer
200 × 23,230	200 × 232	0.5	19	**0.3611**	0.4722	0.6389	0.5833	0.5278
			55	0.3056	0.5556	**0.6944**	0.5833	0.5833
			80	0.3333	**0.5833**	0.6667	**0.7222**	**0.7222**
			99	**0.3611**	0.25	0.5833	0.4444	0.5833
			150	**0.3611**	0.4167	0.6389	0.6667	0.6389
AVERAGE CLASSIFICATION ACCURACY				0.3444	0.45556	**0.6444**	0.59998	0.6111

Appendix shows few clean (cover) images and their embedded counterpart for a specific payload of 0.5bpdct, embedding output, the extracted features stored in .xls file and the convergence of ALO optimizer for 400 iterations. The convergence of the entire ALO for the chosen feature set is shown in Fig. 3.

After optimization the reduced features are classified. The classification accuracy for single and fusion classifiers for different payloads (0.5, 0.8, 0.01, 0.1) are tabulated in Tables 3, 4, 5, and 6 respectively. The highest accuracy values are in bold under each category.

For a payload of 0.5, the highest classification accuracy has been noticed for Decision template (72.22%) and Dempster Schafer (72.22%) fusion classifiers followed by Bayes (69.44%). Considering average accuracy, **Bayes fusion classifier** has the highest **overall average accuracy** (64.44%). When the payload is 0.8, **Bayes** has maximum average accuracy (61.66%) compared to all other classifiers. For payloads of 0.01 and 0.1, again **Bayes classifier has the highest average classification accuracy.** Table 7 shows the performance of all classifiers for different payloads but fixed PRNG value (PRNG seed = 80) during embedding. Even in this case, **Bayes classifier has the highest average classification accuracy.**

Comparing the average classification accuracies of all classifiers in Table 7, Fig. 4 shows that Bayes fusion classifier is the best. Considering the average classification accuracies (from Tables 3, 4, 5, and 6) of all classifiers for different payloads, Fig. 5 shows that Bayes fusion classifier outperforms all other type of classifiers. Thus Bayes fusion classifier can be considered as universal classifier for JPEG steganalysis.

Table 8 shows the timing calculations for different images for a payload of 0.5 and PRNG seed value 80 during embedding. The total time is the sum of the extraction and optimization time of both the cover and stego images along with the embedding time. The average time taken for processing one image is 18.4315 s. Appendix shows the time calculation for stego image

Table 4 Classification accuracy of single and fusion classifiers for Payload = 0.8

Original Feature set size	Optimized Feature set size	Payload during Embedding	PRNG seed during embedding	SVM	MLP	Bayes	Decision Template	Dempster Schafer
200 × 23,230	200 × 232	0.8	19	**0.4167**	0.5278	0.5833	**0.6944**	**0.6944**
			55	0.3056	**0.5833**	0.6667	0.6111	0.5833
			80	0.3333	0.5278	**0.6111**	0.5556	0.5556
			99	0.3611	0.3056	**0.6111**	0.4444	0.5556
			150	0.3611	0.4444	**0.6111**	0.5833	0.6389
AVERAGE CLASSIFICATION ACCURACY				0.3555	0.47778	**0.61666**	0.57776	0.60556

Table 5 Classification accuracy of single and fusion classifiers for Payload = 0.01

Original Feature set size	Optimized Feature set size	Payload during Embedding	PRNG seed during embedding	SVM	MLP	Bayes	Decision Tree	Dempster Schafer
200 × 23,230	200 × 232	0.01	19	**0.4167**	0.3611	0.5278	0.4167	0.4167
			55	0.3056	0.5	0.6111	0.6111	**0.6111**
			80	0.3333	0.4444	**0.6667**	0.6111	**0.6111**
			99	0.3611	**0.5278**	0.6111	0.6111	**0.6111**
			150	0.3611	**0.5278**	0.6389	**0.6389**	0.5833
AVERAGE CLASSIFICATION ACCURACY				0.3555	0.4722	**0.61112**	0.57778	0.56666

(image number 74). Running the algorithm for steganographic embedding, feature extraction and optimization, the time for 200 images was found to be 691.34 s or 11.522 min. This is shown in Appendix. Further the classification of this optimized feature set took 55.09 s. Thus the TOTAL PROCESSING TIME IS 746.43 s or 12.44 min for 200 images.

4 Comparison with prior research

In the recent past, steganalysis with feature extraction has used limited statistical features and neural network based classifiers [13, 23]. While some parallel processing video coding algorithms [3–5] exist for reducing time and space complexity, their application to image processing would be expensive compared to our proposed simple genetic based ALO optimization. Though few research works have concentrated on rich models for universal steganalysis, they use statistical feature reduction techniques and like Fisher Linear Discriminant (FLD) and ensemble classifiers [12]. These ensemble methods take different portions of feature sets for classification and find the Minimal total error (in terms of false alarm and missed detection rate). Table 9 shows the comparison of this research with the earlier research work in this field.

From Table 9 it is obvious that this research has better classification accuracy (highest value of average classification accuracy is shown in bold) than other research works. Comparing with the recent research work in this field, ALO gives better classification accuracy for the most sophisticated nsF5 embedding, while others report steganalysis of HUGO and YASS [26]. Research by Chhikara [7] shows that the feature reduction rate for CCPEV-548 features is 82%. Another research by Chhikara [6] states that the reduction rate is 67% for DCT features and 38% for DWT features. Whereas the feature reduction rate in this research is 99%. This

Table 6 Classification accuracy of single and fusion classifiers for Payload = 0.1

Original Feature set size	Optimized Feature set size	Payload during Embedding	PRNG seed during embedding	SVM	MLP	Bayes	Decision Tree	Dempster Schafer
200 × 23,230	200 × 232	0.1	19	**0.4167**	0.4444	0.5556	0.6111	0.6111
			55	0.3056	**0.5833**	**0.6389**	**0.6389**	**0.6389**
			80	0.3333	0.4722	0.6111	0.5278	0.6389
			99	0.3611	0.3056	0.4722	0.3333	0.3333
			150	0.3611	0.5278	0.6389	0.6389	0.6389
AVERAGE CLASSIFICATION ACCURACY				0.35556	0.4666	**0.58334**	0.55	0.5722

Table 7 Classification accuracy of single and fusion classifiers for different Payloads and fixed PRNG seed

Original Feature set size	Optimized Feature set size	Payload during Embedding	SVM	MLP	Bayes	Decision Tree	Dempster Schafer
200 × 23,230	200 × 232	0.01	0.3333	0.4444	**0.6667**	0.6111	0.6111
		0.1	0.3333	0.4722	0.6111	0.5278	0.6389
		0.5	0.3333	**0.5833**	**0.6667**	**0.7222**	**0.7222**
		0.8	0.3333	0.5278	0.6111	0.5556	0.5556
AVERAGE CLASSIFICATION ACCURACY			0.3333	0.506925	**0.6389**	0.604175	0.63195

high reduction rate enables the use of rich image feature sets (23,230 features) for improved classification accuracy.

While most of the image steganalysis research analyse the results in terms of classification accuracy, research by Kodovsky [12] shows average running time for HUGO, EA and ± Embedding algorithms for their ensemble method. It is seen that for the most simple ± Embedding the running time is 1 h 20 min, for HUGO 4 h 35 min and for EA 3 h 09 min [12]. The time complexity calculation in our research shows 12.44 MINUTES for extracting, optimizing and classifying all 200 images for the sophisticated nsF5 embedding algorithm. Table 10 shows comparison of this research with our earlier research.

The research work presented in this paper gives greater classification accuracy of 64.44% (shown in bold in Table 10) than these earlier methods stated in Tables 8 and 9. Future scope of this research could be fine tuning the parameters of the ALO algorithm or changing the random selection of position of antlions (features) to some other method.

5 Significance of this research

The ALO algorithm guarantees exploration of the entire feature space as it considers the features as random walk of ants. There are only few adjustable parameters in ALO while optimizing. These characteristics justify the use of ALO compared to other nature inspired optimization algorithms. The concept of ALO based JPEG steganalysis is significant compared to related work in this field due to the following reasons,

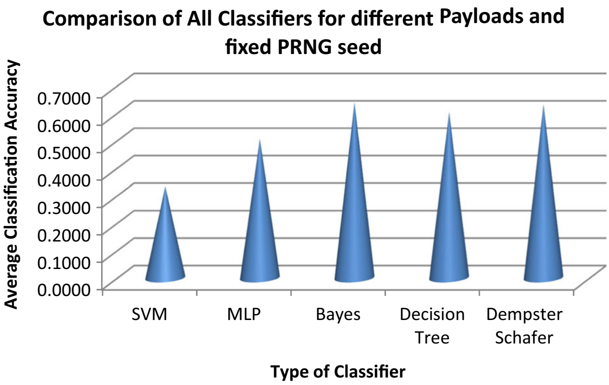


Fig. 4 Performance of all classifiers for different payloads but fixed PRNG value during embedding

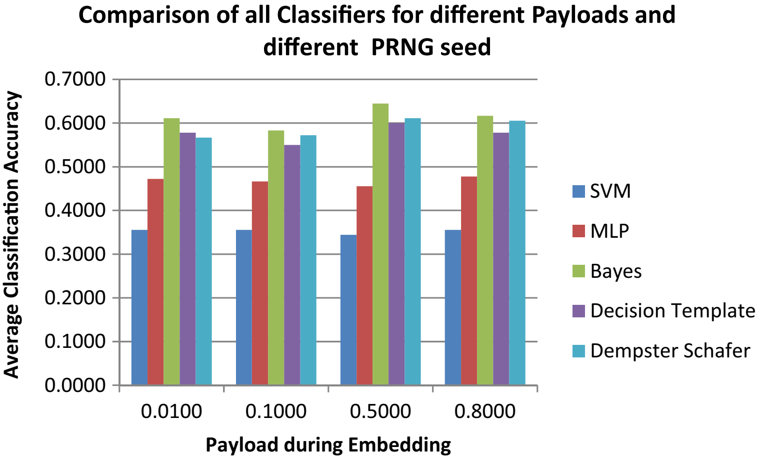


Fig. 5 Average classification accuracies of all classifiers for different payloads averaged for different PRNG seed values

- The feature reduction rate obtained by ALO is far superior compared to other methods [6, 7, 26].
- The method of direct feature optimization and classification used in this research is simpler than the state of art ensemble based feature selection method proposed by Kodovsky and Fridrich [12].
- Compared to the various approaches of feature extraction and steganalysis [14, 20, 23, 29], this approach is not complicated as it considers all possible combinations of feature changes and then optimizes them to give better classification accuracy.

Thus Bayes classifier when used with ALO based optimization gives significant improvement in classification accuracy and reduced time complexity for JPEG steganalysis.

6 Conclusion

This research has implemented a nature inspired meta-heuristic technique for optimizing the high dimensional image features for improved image steganalysis. The steganographic

Table 8 Time Complexity Calculation when payload is 0.5 and PRNG seed = 80

Chosen Image Number	Cover Features		Time for Embedding (seconds)	Number of Embedding Changes	Stego Features		Total Time (seconds)
	Extraction (seconds)	Optimization (seconds)			Extraction (seconds)	Optimization (seconds)	
04	5.31	1.62	0.0367	6370	5.15	6.55	18.6667
25	4.99	1.50	0.0261	2574	5.41	7.14	19.0661
55	5.03	1.50	0.0248	4051	4.79	6.24	17.5848
74	5.26	1.49	0.0258	6195	5.02	6.43	18.4658
98	5.29	1.44	0.0341	6218	5.09	6.52	18.3741
Average Time for processing ONE image							18.4315

Table 9 Comparison with earlier research by other researchers

Steganographic Domain	Researcher(s) name	Type of feature set	Performance measure and it's value	Method of classification	Percentage Error	Percentage Accuracy
JPEG	Shi et al. [27]	648	classification error rate	Single (SVM) Ensemble	43.5% 42.35%	56.5% 57.65%
	Liu [21]	216	classification error rate	Single (SVM) Ensemble	38.71% 36.85%	61.29% 63.15%
	Fridrich and Kodvosky [12]	22,510	Minimal total error (in terms of false alarm and missed detection rate alone)	Ensemble with FLD	0.1616	-
	This research	23,230	Average Percentage Classification Accuracy	ALO Optimization and Single classifiers (SVM, MLP)	-	50.69% for MLP
				ALO Optimization and FUSION classifiers (Bayes, Decision template, Dempster Schafer)	-	**64.44%** for Bayes fusion classifier

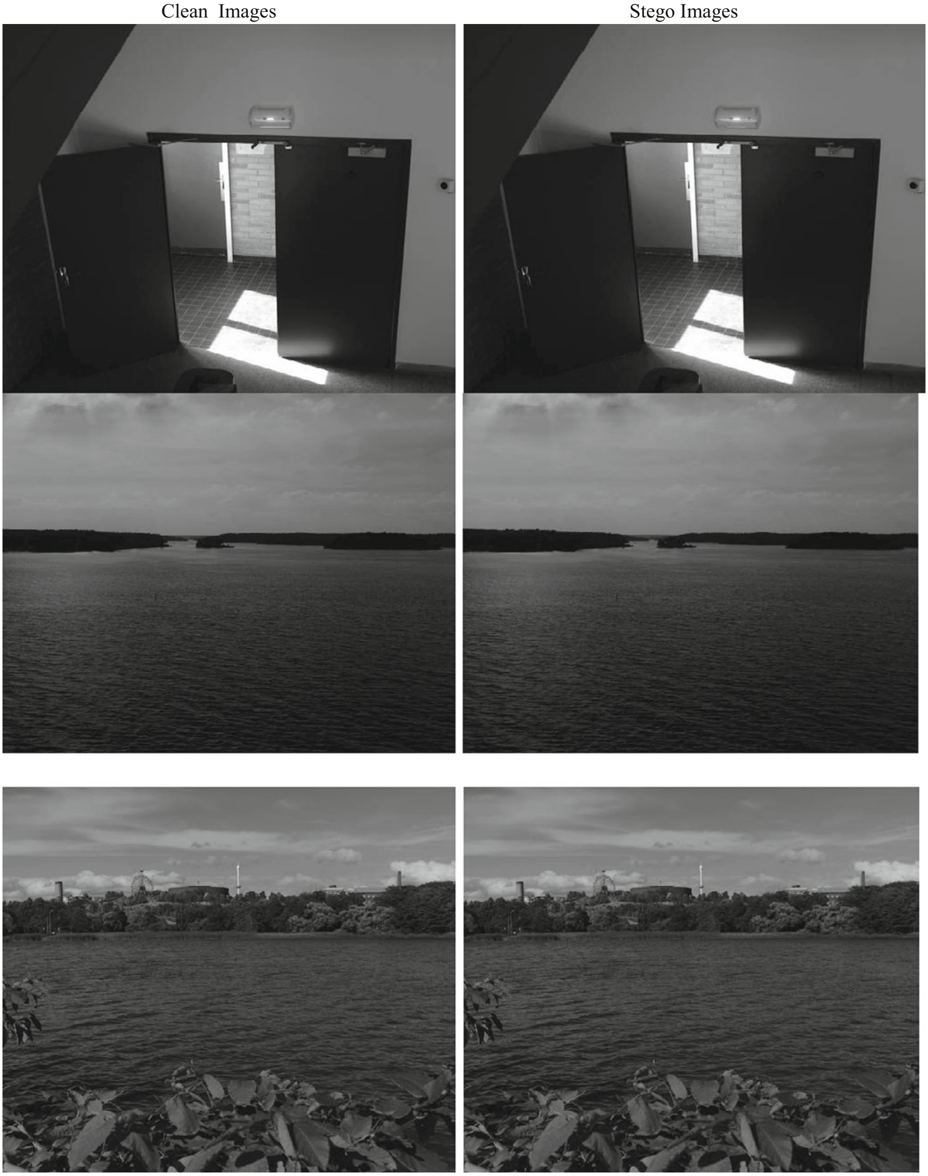
embedding is due to nsF5 algorithm and the extracted features are of dimension 200 × 23,230. The extracted feature model is based on the correlation among the DCT coefficients in frequency and spatial domain. To tackle the problem of computational complexity, the feature set is reduced by Ant Lion Optimization (ALO) technique. In this technique, movements of ants and antlions are considered as random walks in search space and the positions of the ants are the features in iteration. The fitness function is due to the pits built by the antlions. The reduced feature set of dimension 200 × 232 is classified with individual (SVM and MLP) and fusion classifiers (Bayes, Decision template, Dempster Schafer). With classification accuracy and time complexity as the performance measure, results have been analyzed for different payloads and different chosen DCT coefficients (different PRNG seed values). The highest AVERAGE CLASSIFICATION ACCURACY has been noticed for Bayes fusion classifier (64.44%) when the payload is 0.5. It has been noted that the performance of fusion classifiers is good compared to individual classifiers. Hence for JPEG steganalysis, Bayes classifier with ALO based optimization gives better classification accuracy compared to existing research.

Table 10 Comparison with our earlier research

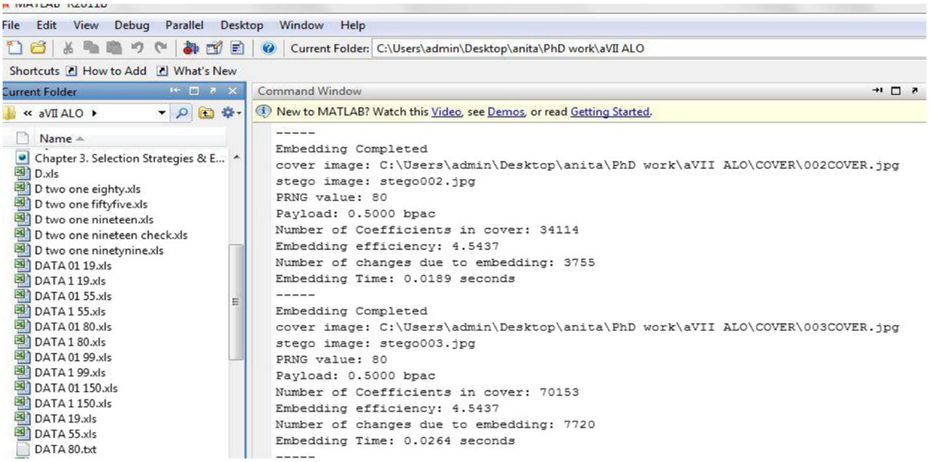
Steganographic Domain	Type of feature set	Type of Optimization technique	Best Classifier	Percentage Accuracy
JPEG	548 [8]	nil	Single - SVM	61.1%
	8726 [10]	Clustering	Single - SVM Fusion – Bayes	41.26% 58.74%
	48,600 [11]	ALO	Single - MLP Fusion - Bayes	46.11% 60.55%
	23,230 (this research)	ALO	Single - MLP Fusion - Bayes	50.69% **64.44%**

Appendices

A. Few of the Clean and Stego images in this research for Payload = 0.5 bpdct

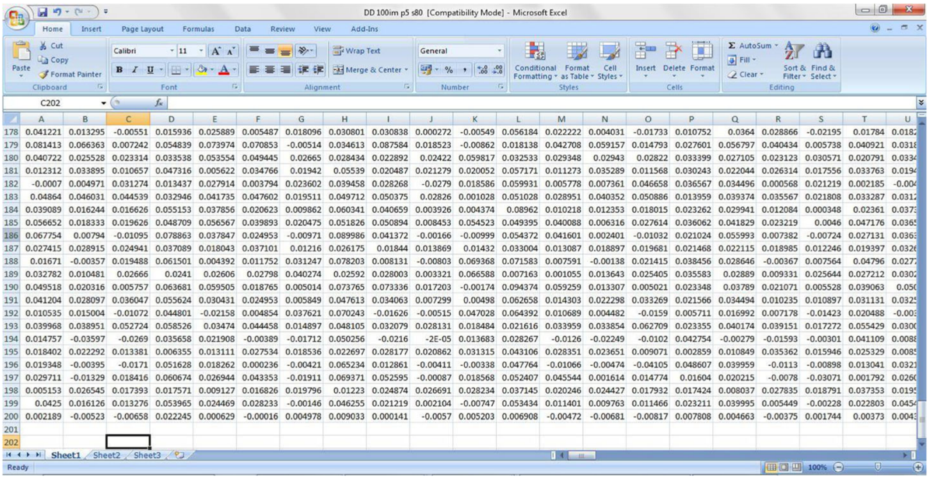


Embedding output



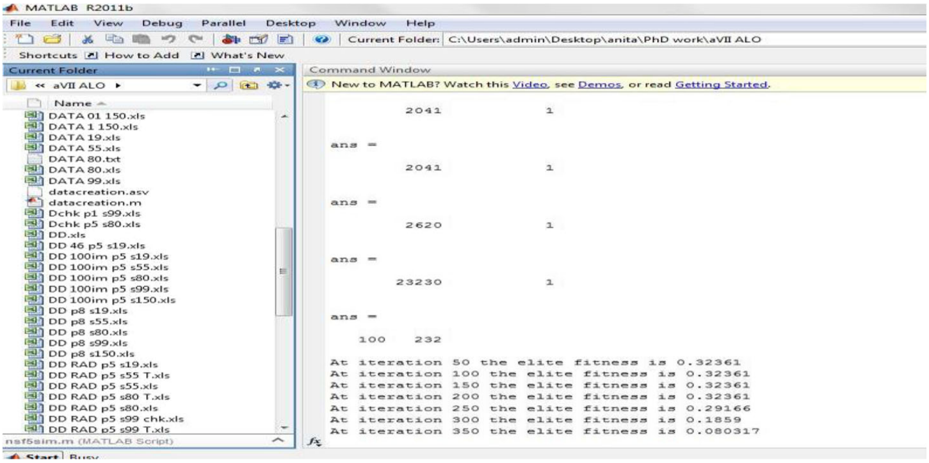
```
-----
Embedding Completed
cover image: C:\Users\admin\Desktop\anita\PhD work\aVII ALO\COVER\002COVER.jpg
stego image: stego002.jpg
PRNG value: 80
Payload: 0.5000 bpac
Number of Coefficients in cover: 34114
Embedding efficiency: 4.5437
Number of changes due to embedding: 3755
Embedding Time: 0.0189 seconds
-----
Embedding Completed
cover image: C:\Users\admin\Desktop\anita\PhD work\aVII ALO\COVER\003COVER.jpg
stego image: stego003.jpg
PRNG value: 80
Payload: 0.5000 bpac
Number of Coefficients in cover: 70153
Embedding efficiency: 4.5437
Number of changes due to embedding: 7720
Embedding Time: 0.0264 seconds
-----
```

B. Features extracted from 200 images (100 clean and 100 stego) for payload of 0.5.


DD 100im p5 s80 [Compatibility Mode] - Microsoft Excel
Home Insert Page Layout Formulas Data Review View Add-Ins
Paste Cut Copy Format Painter Clipboard
Calibri 11 Font
Wrap Text Merge & Center Alignment
General Number
Conditional Formatting Format as Table Cell Styles Styles
Insert Delete Format Cells
AutoSum Fill Clear Sort & Filter Find & Select Editing
C202
Sheet1 Sheet2 Sheet3
Ready
100%

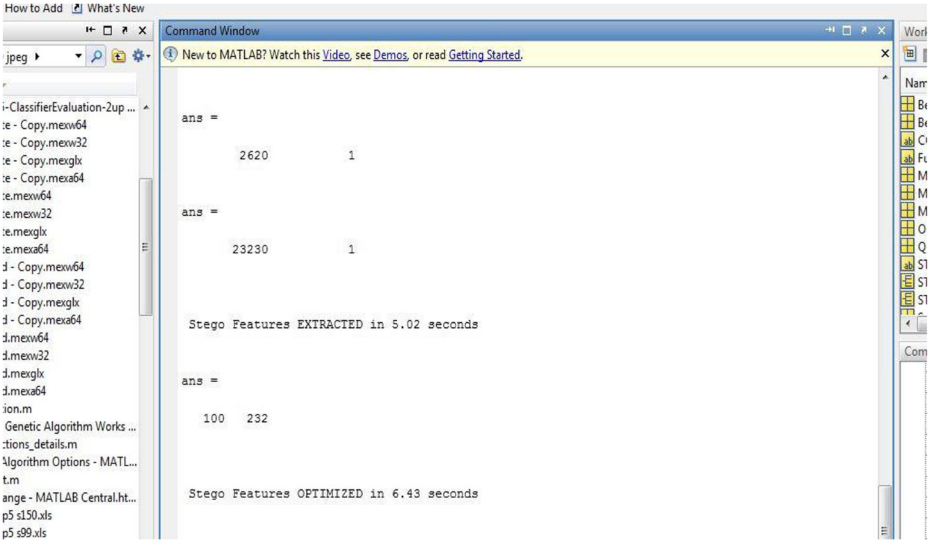
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
178	0.041221	0.013295	-0.00551	0.015936	0.025889	0.005487	0.018096	0.030801	0.030838	0.000272	-0.00549	0.056184	0.022222	0.004031	-0.01733	0.010752	0.0364	0.028866	-0.02195	0.01784	0.0182
179	0.081413	0.066363	0.007242	0.054839	0.073974	0.070853	-0.00514	0.034613	0.087584	0.018523	-0.00862	0.018138	0.042708	0.059157	0.014793	0.027601	0.056797	0.040434	0.005738	0.040921	0.0318
180	0.040722	0.025528	0.023314	0.033538	0.053554	0.049445	0.02665	0.028434	0.022892	0.02422	0.059817	0.032533	0.029348	0.02943	0.02822	0.033399	0.027105	0.023123	0.030571	0.020791	0.0334
181	0.012312	0.033895	0.010657	0.047316	0.005622	0.034766	0.01942	0.05539	0.020487	0.021279	0.020052	0.057171	0.011273	0.035289	0.011568	0.030243	0.022044	0.026314	0.017556	0.033763	0.0194
182	-0.0007	0.004971	0.031274	0.013437	0.027914	0.003794	0.023602	0.039458	0.028268	-0.0279	0.018586	0.059931	0.005778	0.007361	0.046658	0.036567	0.034496	0.000568	0.021219	0.002185	-0.004
183	0.04864	0.046031	0.044539	0.032946	0.041735	0.047602	0.019511	0.049712	0.050375	0.02826	0.001028	0.051028	0.028951	0.040352	0.050886	0.013959	0.039374	0.035567	0.021808	0.033287	0.0312
184	0.039089	0.016244	0.016626	0.055153	0.037856	0.020623	0.009862	0.060341	0.040659	0.003926	0.004374	0.08962	0.010218	0.012353	0.018015	0.023262	0.029941	0.012084	0.000348	0.02361	0.0373
185	0.056652	0.018333	0.019626	0.048709	0.056567	0.039893	0.020475	0.051826	0.050894	0.008453	0.054523	0.049395	0.040088	0.006316	0.027614	0.036062	0.041829	0.023219	0.0046	0.047176	0.0365
186	0.067754	0.00794	-0.01095	0.078863	0.037847	0.024953	-0.00971	0.089986	0.041372	-0.00166	-0.00999	0.054372	0.041601	0.002401	-0.01032	0.021024	0.055993	0.007382	-0.00724	0.027131	0.0363
187	0.027415	0.028915	0.024941	0.037089	0.018043	0.037101	0.01216	0.026175	0.01844	0.013869	0.01432	0.033004	0.013087	0.018897	0.019681	0.021468	0.022115	0.018985	0.012246	0.019397	0.0326
188	0.01671	-0.00357	0.019488	0.061501	0.004392	0.011752	0.031247	0.078203	0.008131	-0.00803	0.069368	0.071583	0.007591	-0.00138	0.021415	0.038456	0.028646	-0.00367	0.007564	0.04796	0.0277
189	0.032782	0.010481	0.02666	0.0241	0.02606	0.02798	0.040274	0.02592	0.028003	0.003321	0.066588	0.007163	0.001055	0.013643	0.025405	0.035583	0.02889	0.009331	0.025644	0.027212	0.0302
190	0.049518	0.020316	0.005757	0.063681	0.059505	0.018765	0.005014	0.073765	0.073336	0.017203	-0.00174	0.094374	0.059259	0.013307	0.005021	0.023348	0.03789	0.021071	0.005528	0.039063	0.050
191	0.041204	0.028097	0.036047	0.055624	0.030431	0.024953	0.005849	0.047613	0.034063	0.007299	0.00498	0.062658	0.014303	0.022298	0.033269	0.021566	0.034494	0.010235	0.010897	0.031131	0.0325
192	0.010535	0.015004	-0.01072	0.044801	-0.02158	0.004854	0.037621	0.070243	-0.01626	-0.00515	0.047028	0.064392	0.010689	0.004482	-0.0159	0.005711	0.016992	0.007178	-0.01423	0.020488	-0.003
193	0.039968	0.038951	0.052724	0.058526	0.03474	0.044458	0.014897	0.048105	0.032079	0.028131	0.018484	0.021616	0.033959	0.033854	0.062709	0.023355	0.040174	0.039151	0.017272	0.055429	0.0300
194	0.014757	-0.03597	-0.0269	0.035658	0.021908	-0.00389	-0.01712	0.050256	-0.0216	-2E-05	0.013683	0.028267	-0.0126	-0.02249	-0.0102	0.042754	-0.00279	-0.01593	-0.00301	0.041109	0.0088
195	0.018402	0.022292	0.013381	0.006355	0.013111	0.027534	0.018536	0.022697	0.028177	0.020862	0.031315	0.043106	0.028351	0.023651	0.009071	0.002859	0.010849	0.035362	0.015946	0.025329	0.0085
196	0.019348	-0.00395	-0.0171	0.051628	0.018262	0.000236	-0.00421	0.065234	0.012861	-0.00411	-0.00338	0.047764	-0.01066	-0.00474	-0.04105	0.048607	0.039959	-0.0113	-0.00898	0.013041	0.0321
197	0.029711	-0.01329	0.018416	0.060674	0.026944	0.043353	-0.01911	0.069371	0.052595	-0.00087	0.018568	0.052407	0.045544	0.001614	0.014774	0.01604	0.020215	-0.0078	-0.03071	0.001792	0.0260
198	0.005153	0.026545	0.017393	0.017571	0.009127	0.016826	0.019796	0.01223	0.024874	0.026691	0.028234	0.037145	0.020246	0.024427	0.017932	0.017424	0.008037	0.027835	0.018791	0.037353	0.0195
199	0.0425	0.016126	0.013276	0.053965	0.024469	0.028233	-0.00146	0.046255	0.021219	0.002104	-0.00747	0.053434	0.011401	0.009763	0.011466	0.023211	0.039995	0.005449	-0.00228	0.022803	0.0454
200	0.002189	-0.00523	-0.00658	0.022245	0.000629	-0.00016	0.004978	0.009033	0.000141	-0.0057	0.005203	0.006908	-0.00472	-0.00681	-0.00817	0.007808	0.004663	-0.00375	0.001744	0.00373	0.0043
201																					
202																					

C. Convergence of Elite fitness by ALO optimizer for 400 iterations



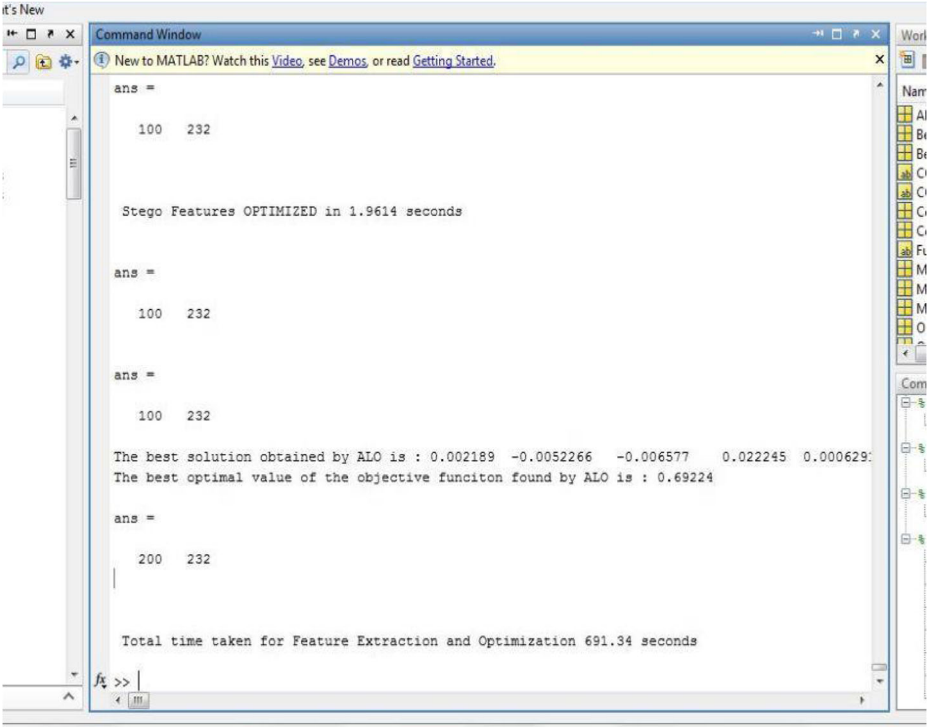
```
        2041            1


ans =

        2041            1


ans =

        2620            1


ans =

       23230            1


ans =

   100   232

At iteration 50 the elite fitness is 0.32361
At iteration 100 the elite fitness is 0.32361
At iteration 150 the elite fitness is 0.32361
At iteration 200 the elite fitness is 0.32361
At iteration 250 the elite fitness is 0.29166
At iteration 300 the elite fitness is 0.1859
At iteration 350 the elite fitness is 0.080317
```

D. Time Complexity Calculation for Image number 74



```
ans =

       2620            1


ans =

      23230            1



 Stego Features EXTRACTED in 5.02 seconds


ans =

   100   232



 Stego Features OPTIMIZED in 6.43 seconds
```

E. Time Calculation for 200 images (Feature extraction and optimization)



New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
ans =

   100   232



 Stego Features OPTIMIZED in 1.9614 seconds


ans =

   100   232


ans =

   100   232

The best solution obtained by ALO is : 0.002189  -0.0052266   -0.006577    0.022245  0.000629:
The best optimal value of the objective funciton found by ALO is : 0.69224

ans =

   200   232



 Total time taken for Feature Extraction and Optimization 691.34 seconds

fx >>
```

References

1. Bas P, Filler T, Pevny T (2011) Break our steganographic system — the ins and outs of organizing BOSS. In: Proceedings of Information Hiding Conference 6958:59–70. http://dde.binghamton.edu/download/. Accessed 2 May 2016
2. Bianchi L, Dorigo M, Gambardella LM, Gutjahr WJ (2009) A survey on metaheuristics for stochastic combinatorial optimization. Nat Comput 8:239–287 http://code.ulb.ac.be/dbfiles/BiaDorGamGut2009natcomp.pdf. Accessed 30 May 2016
3. Chenggang Y, Yongdong Z, Jizheng X, Feng D, Liang L, Qionghai D, Feng W (2014) A highly parallel framework for HEVC coding unit partitioning Tree decision on many-core processors. IEEE Signal Processing Lett 21(5):573–576
4. Chenggang Y, Yongdong Z, Jizheng X, Feng D, Jun Z, Qionghai D, Feng W (2014) Efficient parallel framework for HEVC motion estimation on many-Core processors. IEEE Trans Circuits Syst Video Technol 24(12):2077–2089
5. Chenggang Y, Yongdong Z, Feng D, Jizheng X, Liang L, Qionghai D (2014) Efficient parallel HEVC intra prediction on many-core processor. Electron Lett 50(11):805–806
6. Chhikara RR, Sharma P, Singh L (2016) An improved dynamic discrete firefly algorithm for blind image steganalysis. Int J Mach Learn Cybern. doi:10.1007/s13042-016-0610-3
7. Chikkara RR, Singh L (2017) An improved discrete firefly and t-test based algorithm for blind image steganalysis, In: Proc. of 6th International Conference on Intelligent Systems, Modelling and Simulation (ISMS). IEEEXplore Digital library. http://ieeexplore.ieee.org/document/7311210/. Accessed May 24

8. Christaline JA, Ramesh R, Vaishali D (2014) Steganalysis with classifier combinations. ARPN Journal of Engineering and Applied Sciences 9(12). http://www.arpnjournals.com/jeas/research_papers/rp_2014/jeas_1214_1402.pdf. Accessed 2 May 2016
9. Christaline JA, Ramesh R, Vaishali D (2015) Critical review of image steganalysis techniques. International Journal of Advanced Intelligence Paradigms, Inderscience 7(3/4):368–381 www.inderscienceonline.com/doi/abs/10.1504/IJAIP.2015.073715. Accessed 27 May 2016
10. Christaline JA, Ramesh R, Vaishali D (2016) Optimized JPEG Steganalysis. International Journal of Multimedia and Ubiquitous Engineering, SERSC 11(1):385–396. doi:10.14257/ijmue.2016.11.1.37 Accessed 2 May 2016
11. Christaline JA, Ramesh R, Vaishali D (2016) Bio-inspired computational algorithms for improved image steganalysis. Indian Journal of Science and Technology 9(10). http://www.indjst.org/index.php/indjst/article/viewFile/88995/68459. Accessed 2 May 2016
12. Fridrich J, Kodvosky J (2012) Rich models for Steganalysis of digital images. IEEE Trans Inf Forensics Secur 7(3):868–882 http://dde.binghamton.edu/kodovsky/pdf/TIFS2012-SRM.pdf. Accessed 30 May 2016
13. Holub V, Fridrich J (2013) Random projection s of residuals for digital image Steganalysis. IEEE Trans Inf Forensics Secur 8(12):1996–2006 http://dde.binghamton.edu/vholub/pdf/TIFS13_Random_Projections_of_Residuals_for_Digital.pdf. Accessed 22 May 2016
14. Huang F, Li B, Huang J (2008) Universal JPEG steganalysis based on microscopic and macroscopic calibration. In: Proceedings IEEE International Conference on Image Processing ICIP, 52:2068–2071. http://ieeexplore.ieee.org/document/4712193/. Accessed 19 May 2016
15. Kaveh A, Khayatazad M (2012) A new meta-heuristic method: ray optimization. Comput Struct 112:283–294 www.sciencedirect.com/science/article/pii/S0045794912002131. Accessed 29 May 2016
16. Kaveh A, Ghazaan MI, Bakhshpoori T (2013) An improved ray optimization algorithm for design of truss structures. Civ Eng 57:97–112 https://pp.bme.hu/ci/article/viewFile/7166/6159. Accessed 31 May 2016
17. Kaveh A, Ghazaan MI, Bakhshpoori T (2013) An improved ray optimization algorithm for design of truss structures. Civ Eng 57:97–112. https://pp.bme.hu/ci/article/viewFile/7166/6159. Accessed 1 June 2016
18. Kodovsky J, Fridrich, J (2011) Steganalysis in high dimension: fusing classifiers built on random subspace. In: Proc. Of SPIE, Electronic Imaging, Media, Watermarking, Security and Forensics XIII, pp. 23–26. http://dde.binghamton.edu/kodovsky/pdf/Kod11spie.pdf. Accessed 21 May 2016
19. Kuncheva LI (2002) A theoretical study on six classifier fusion strategies. IEEE Trans Pattern Anal Mach Intell 24(2):281–286 http://machine-learning.martinsewell.com/ensembles/Kuncheva2002a.pdf. Accessed 2 May 2016
20. Liu Q (2011) Detection of misaligned cropping and recompression with the same quantization matrix and relevant forgery. In: Proceedings of the 3rd international ACM workshop on Multimedia in forensics and intelligence, pp 25–30. http://www.shsu.edu/~qxl005/New/Publications/Mifor2011.pdf. Accessed 29 May 2016
21. Liu Q (2011) Steganalysis of DCT embedding based adaptive steganography and YASS. In: Proceedings of the 13th ACM workshop on Multimedia and Security, Buffalo. https://pdfs.semanticscholar.org/ca6f/92a55790769a56fe9c4b87179a09fd50df9a.pdf. Accessed 22 May 2016
22. Mirjalili SA (2015) The ant Lion optimizer. Adv Eng Softw 83:80–98. doi:10.1016/j.advengsoft.2015.01 Accessed 28 May 2016
23. Pevny T, Fridrich J (2007) Merging Markov and DCT features for multi-class JPEG steganalysis. In: Delp EJ, Wong PW (eds) Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, 6505(1):3–14. https://pdfs.semanticscholar.org/b7a7/700eaf1c2803a511ac4ca71ae128e09b2a18.pdf. Accessed 29 May 2016
24. Rajabioun R (2011) Cuckoo Optimization Algorithm. Applied Soft Computing 11:5508–5518. dl.acm.org/citation.cfm?id=2039522. Accessed 31 May 2016
25. Roshini D, Samsudin A (2009) A digital steganalysis: computational intelligence approach. Int J Comput 3(1):161–170 https://www.researchgate.net/profile/Roshidi_Din/publication/228844865_Digital_Steganalysis_Computational_Intelligence_Approach/links/0fcfd5066a469e19ac000000.pdf?origin=publication_list. Accessed 25 May 2016
26. Sajedi H (2017) Image steganalysis using artificial bee colony algorithm. J Exp Theor Artif Intell. http://tandfonline.com/doi/abs/10.1080/0952813X.2016.1266037. Accessed 7 June 2017
27. Shi YQ, Chen C, and Chen W (2006) A Markov process based approach to effective attacking JPEG steganography. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.646.5540&rep=rep1&type=pdf. Accessed 22 May 2016
28. Talbi EG (2009) Metaheuristics: from design to implementation, vol 74. Sons, John Wiley &
29. Wang Y, Liu JF, Zhang WM (2009) Blind JPEG steganalysis based on correlations of DCT coefficients in multi-directions and calibrations. In: Proceedings of the 2009 International Conference on Multimedia Information Networking and Security, 1:495–499. http://ieeexplore.ieee.org/document/5368437/. Accessed 9 June 2016

30. Windeatt T (2006) Accuracy/diversity and ensemble MLP classifier design. IEEE Trans Neural Netw 17(5): 1194–1211 http://ieeexplore.ieee.org/document/1687930/. Accessed 12 May 2016
31. Yang XS (2010) Firefly algorithm, stochastic test functions and design optimization. International Journal of Bio-Inspired Computation 2(2):78–84 https://arxiv.org/pdf/1003.1409.pdf. Accessed 1 June 2016
32. Yang XS, Deb S (2010) Engineering optimisation by cuckoo search. International Journal of Mathematical Modelling and Numerical Optimization 1(4):330–343

J. Anita Christaline completed her B.E. Degree in Electronics and Communication Engineering in 1996 from Bharathidasan University, Trichy. She is recipient of All India Exam - GATE (Graduate Aptitude Test in Engineering) during 2003 and secured high score. She received her M.E. Degree in Applied Electronics in the year 2006 from Anna University. She is currently working as Assistant Professor in SRM University, Chennai where she is pursuing her Doctoral Degree. Her area of interest is Image Steganography and Steganalysis by computational intelligence. She has published many papers in this field. She is member IEEE and IET.

R. Ramesh was born in Kanyakumari, India, 1976. He received B.E.Degree in Electronics and Communication Engineering in 1998 and M.E. degree in Communication Systems in 2000, both from Madurai Kamaraj University, India. He has been awarded Doctoral degree in SRM University in the year 2009 for his research work on Testing the Stability of two dimensional recursive filters. He is currently working as a Professor in the Department of Electronics and Communication Engineering at Saveetha Engineering College, Chennai. India. His current research interests concern digital image and signal processing.

C. Gomathy acquired a B.E.degree in Electronics and Communication Engineering from Government College of Engineering, Tirunelveli, in the year 1986, and an M.S. degree in Electronics and Control Engineering from BITS, Pilani, in 1992. She also obtained an M.S. degree from Anna University in 2001. She obtained her Ph.D. in the area of Mobile Ad hoc networks from College of Engineering, Anna University, Chennai, India, in the year 2007. She has published over 70 research papers in National and International conferences and journals. Her areas of interest include mobile ad hoc networks, high-speed networks, and wireless sensor networks.

D.Vaishali was born in Pune, India, 1969. She received B.E. Degree in Electronics and Telecommunication Engineering in 1994 and the M.E. degree in Communication Systems in 2002, both from Pune University, India. She is a research scholar in the SRM University and her research work is progressing in the field of Image Processing with wavelet Transforms. She is currently working as a Asst. Professor in the Department of Electronics and Communication Engineering at SRM University, Chennai. India. She is member IEEE and IET.