Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing

Pourya Hoseini [a,*], Mahrokh G. Shayesteh [a,b]

[a] *Department of Electrical Engineering, Urmia University, Urmia, Iran*
[b] *Wireless Research Lab., ACRI, Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran*

Keywords:
Image processing
Contrast enhancement
Ant Colony Optimisation (ACO)
Genetic Algorithm (GA)
Simulated Annealing (SA)
Hybrid metaheuristics

A B S T R A C T

In this paper, we propose a hybrid algorithm including Genetic Algorithm (GA), Ant Colony Optimisation (ACO), and Simulated Annealing (SA) metaheuristics for increasing the contrast of images. In this way, contrast enhancement is obtained by global transformation of the input intensities. Ant colony optimisation is used to generate the transfer functions which map the input intensities to the output intensities. Simulated annealing as a local search method is utilised to modify the transfer functions generated by ant colony optimisation. And genetic algorithm has the responsibility of evolutionary process of ants' characteristics. The employed fitness function operates automatically and tends to provide a balance between contrast and naturalness of images. The results indicate that the new method achieves images with higher contrast than the previously presented methods from the subjective and objective viewpoints. Further, the proposed algorithm preserves the natural look of input images.

1. Introduction

Contrast enhancement is one of the key steps in image enhancement. It is used for expanding the range of intensities in a grey scale image, revealing parts of image which are seen difficult, or changing the histogram of images so that they can be interpreted more easily by human. In addition to improve the visual perception of an image, contrast increasing is a common preprocessing stage in many image processing applications.

One of the simplest and popular ways to perform contrast enhancement is global intensity transformation. In this method, by utilising lookup tables, the intensity levels in an image are mapped into a new set of grey levels thus changing the image parameters like the contrast. It is called "global" since it performs intensity transformation on all grey levels of an image not on a group of them. The main objective in global intensity transformation is to obtain a lookup table or transfer function which yields an output image with improvement in desired parameters. In the case of contrast enhancement, the desired transfer function should map intensity levels in a way that the contrast of image increases while its natural look is preserved.

Metaheuristics like Genetic Algorithm (GA), Simulated Annealing (SA), and Ant Colony Optimisation (ACO) are guided random searches to find an optimum solution usually in large search spaces (because of difficulties that the classic methods may have in large search spaces). Genetic algorithm, a natural evolutionary method is a global search method. Compared to ACO and SA, GA has the advantage of faster convergence. Ant colony optimisation is based on the natural pheromone trail deposition and attraction of ants when they seek for food. It uses positive feedback for quick find of good solutions [1]. Simulated annealing is originated from the heating and then slow cooling in metallurgy annealing process. Although SA searches locally, but in high temperatures it can select worse solutions so that it can escape local optima. SA has the property of a global metaheuristic.

Recently, hybridisation of metaheuristics has been received great interest. Simulated annealing is often used as a local search in combination with other methods. Various combinations of ant colony optimisation and genetic algorithm have been proposed. In [1], GA sets the controlling parameters of ACO. In [2], ACO is used as a local search for GA. In [3], GA and ACO run in parallel and then the best solution of them is selected.

One of the interesting applications of metaheuristics is in image processing. Ant colony optimisation is used in [4] and [5] for edge detection. In [6], [7], [8], and [9] ACO is applied for interpolation, denoising, thresholding, and segmentation of images, respectively. In [10] and [11], GA is used for global transformation of intensities. In [12] and [13], local image enhancement is performed by GA and particle swarm optimisation.

In this paper, we use global contrast enhancement in the sense of intensity transformation of grey scale images. Transfer function, which maps the input intensities to the output intensities,

* Corresponding author.
E-mail addresses: st_p.hoseini@urmia.ac.ir (P. Hoseini), m.shayesteh@urmia.ac.ir (M.G. Shayesteh).

can have large number of shapes. Hence, metaheuristics are good candidates to perform optimisation in this large search space. We propose a hybrid method which is a combination of GA, ACO, and SA to achieve better results. In the new method, ACO generates the transfer functions considering directions of ants' movement. The characteristics and moving directions of ants are set by GA automatically. This causes GA generally affect the search process since it is able to quickly find the correct global optimum. Also, SA as a local search modifies and enhances the transfer functions and pheromone traces generated by ACO. Our proposed fitness criterion does not need subjective evaluation of users [10,14] and it has no external parameters, i.e. it is not case dependent and not determined by user [11]. We compare the new method with the previously presented methods. The experimental results show that our method outperforms the other ones from the subjective and objective viewpoints.

The rest of the paper is organised as follows. In Section 2, the proposed hybrid method is explained. In Section 3, we present the experimental results and compare with several common methods. Finally, Section 4 concludes the paper.

2. Proposed contrast enhancement mechanism

Considering the transfer function (mapping function) of a grey scale image, one can imagine it as a search space. For an 8-bit image, the mapping function can be assumed to be a 256×256 search pane. In this way, by using ant colony optimisation we can put several ants to walk through the search pane. Reached to the end point, each ant creates a mapping function. In the proposed method, ants' genetic properties are changed as the algorithm proceeds. In this manner, genetic algorithm has the responsibility of modifying the characteristics of artificial ants by adjusting their artificial chromosomes.

At the local search phase, simulated annealing performs fine tuning of the generated transfer functions. The idea for collaborating simulated annealing in the algorithm is that we can assume each point of a transfer function as an atom. Based on the SA method which will be described in Subsection 2.4, the selected atoms and their neighbours may be displaced from their locations causing slight change in transfer functions. By these replacements SA generally improves output mapping functions.

In the new method, by mapping the input intensities to the output intensities according to a transfer function, low contrast images are converted to high contrast ones. The range of intensities in a grey scale image is between 0 and 255. Hence, the outputs of transfer functions generated by ACO and SA have the same range. Fig. 1 shows a typical transfer function. In the proposed method all the metaheuristics generate or preserve monotonically increasing of mapping functions. In the following, we first explain the proposed hybrid algorithm thoroughly and then describe different parts of the method separately.

2.1. Hybrid algorithm

The flowchart of the hybridisation plan is illustrated in Fig. 2. As shown, each individual in GA sets the parameters of two ants in ACO. Then, ants generate transfer functions. In the SA phase, a predefined number of transfer functions obtained in the last run of ACO are selected randomly. Then, some random points in each transfer function are chosen. Finally, SA optimises each selected point and its neighbours for a specified number of cycles. Hybridisation is achieved by performing GA and SA in some of the ACO iterations. GA controls the parameters of ACO and moving directions of ants, hence it needs to be run several times at the first stages of algorithm in order to reach the near optimum solution.

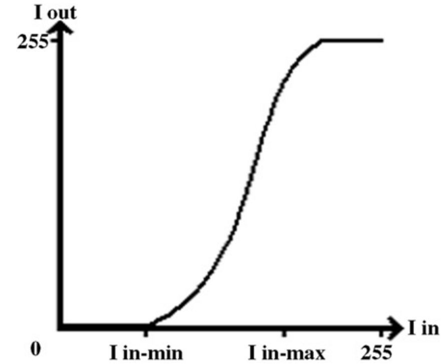


Fig. 1. A typical transfer function.

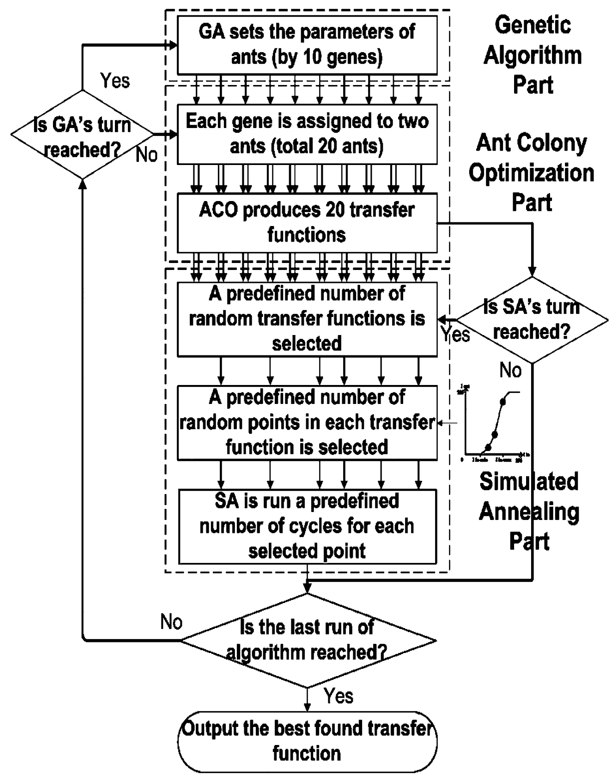


Fig. 2. Flowchart of the proposed algorithm.

SA, as a local search, is run frequently at the last stages of algorithm to yield better solutions and to optimise pheromone trails. In addition, in SA, the number of transfer functions, the points of each transfer function to be optimised, and the number of optimisation process cycles are increased in the last stages. Fig. 3 shows the execution process of GA and SA in a thorough run of the hybrid algorithm. In this figure, each line denotes a predefined GA/SA turn. In order to have the same processing time in different runs of algorithm, the stopping condition is set to be a predefined number of ACO iterations. Here, we consider 100 iterations which was empirically found to reach the desired contrast enhancement while maintains the calculation time reasonable.

Considering the characteristics of evolutionary algorithms (like GA, SA, and ACO), the proposed method has rather many parameters as expected. Those parameter values which empirically yield rather good results in the tests have been used as default parameters. However, in some special cases, the properties of evolutionary algorithms may be altered by changing the default values, based on the need. This ensures the flexibility of the proposed method. The aforementioned generalisation and flexibility subjects stand

Fig. 3. GA and SA schedule in the proposed hybrid algorithm.

Table 1
Parameters of GA and SA executions at different stages of hybrid algorithm schedule.

Phase	Iterations between GA executions	Iterations between SA executions	SA transfer functions to be optimised	SA: Points to be optimised in each transfer function	SA cycles (optimisation duration) in every run
1	5	10	1	2	3
2	6	6	2	4	6
3	7	3	4	6	12

for total hybrid algorithm. For the case of termination condition, a fixed iteration number method is selected and by default is set to 100. Like most parameters of evolutionary algorithms this parameter was chosen based on the result obtained by trial and error. Anyway, the user may change the iteration number if he finds it insufficient or redundant. Also regarding the execution time of GA and SA in the hybrid algorithm (Fig. 3), it should be noted that they depend on the iteration number of algorithm (so user does not need to change them manually), meaning that the predefined schedules are modified and adopted by the program (based on the hybrid algorithm length). However, in case of changing iteration number by user, the resulted schedules will have the properties as presented in Fig. 3, i.e. more SA runs and less GA runs at the last stages of hybrid algorithm. In fact the proposed execution procedure consists of three phases, where the run of each phase constitutes one third of the iteration number. Table 1 demonstrates the values assigned to each parameter of the schedule. In one third of run of the algorithm, GA is run more frequently while SA is less active (i.e. low frequency run, less selected transfer functions, less number of points in each transfer function to be optimised, and less optimisation cycles). In the second stage, GA and SA runs are rather the same. Finally in the last stage, SA is run more and has major impact on the results, while GA less affects the process.

In the following pseudo code, different stages of the proposed hybrid algorithm are described. As indicated, when SA is run, it modifies the acquired solutions of ACO and changes ants' pheromone trails. Also pheromone updating is performed at the end of iterations. Therefore, at the iterations, in which SA is run, the modified results of ACO change the current pheromone map in the update stage.

```
RESET all variables to 0
GENERATE initial population of GA
INITIALIZE temperature
WHILE (iteration<max_iteration_number)
    COMPUTE ACO (ants generate transfer functions)
    COMPUTE fitness of generated transfer functions
    IF (any_of_new_fitnesses>best_fitness) THEN
      SET new_fitness to best_fitness
      SET new pheromone trail to best pheromone trail
      SET new transfer function to best transfer function
    ENDIF
    IF it is SA's turn THEN
      FOR (some_of_ACO's_last_run_transfer_functions)
        FOR (some_points_of_selected_transfer_function)
          FOR (a specified number of cycles)
            EDIT selected transfer function
            CALCULATE fitness of new transfer function
            DECIDE to replace previous transfer function and corresponding pheromone trail and fitness with new ones based on P(old_fitness, new_fitness, temperature)
          ENDFOR
        ENDFOR
      ENDFOR
      IF (any_of_new_fitnesses>best_fitness) THEN
        SET new_fitness to best_fitness
        SET new pheromone trail to best pheromone trail
        SET new transfer function to best transfer function
      ENDIF
      DECREASE temperature
    ENDIF
    IF it is GA's turn THEN
      CALCULATE fitness of each individual
      IF (fitness_of_each_individual>GA_best_fitness) THEN
          SET new GA fitness to GA best fitness
          SET new chromosome to GA best chromosome
      ENDIF
      PROCESS Selection
      PROCESS Reproduction
      PROCESS Replacement
      IF it is last turn of GA THEN
          SET best chromosome to parameters of all ants
      ENDIF
    ENDIF
    COMPUTE pheromone update
  ENDWHILE
RETURN best found transfer function as output.
```

2.2. Ant colony optimisation part

Ant colony optimisation is established on the movement of artificial ants in the search space. Each ant deposits a substance called pheromone when searching. More good results achieved by an ant lead to stronger pheromone deposition. Pheromone causes that other ants be disposed to choose previous ant's way with more probability. Therefore ACO uses the convergence property of pheromone trails [15] while pheromone rate and its deviation determine the expected convergence time [16]. On the other side, evaporation of pheromone assures exploration and tries to prevent immature convergences.

Ant colony optimisation is the core of the hybrid algorithm. It achieves near optimum transfer functions. The basic component of ACO is the dynamic evaluation value (pheromone) which affects ant's decision for moving. Solution construction and pheromone update are two common stages in ACO iterations. In addition to

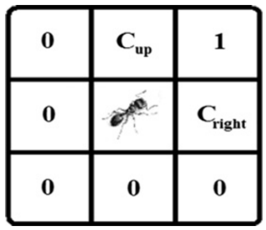


Fig. 4. Heuristic values of ant's neighbourhood.

pheromone, the static evaluation value (heuristic value) is another effective parameter in ant's decision.

Generation of mapping functions is performed by ants' movement from the point $I_{in} = I_{in\text{-}min}$, $I_{out} = 0$ (starting point) to the last point $I_{in} = I_{in\text{-}max}$, $I_{out} = 255$ in a 256×256 search space, as shown in Fig. 1. Note that, points with $I_{in} < I_{in\text{-}min}$ are mapped to $I_{out} = 0$ and points with $I_{in} > I_{in\text{-}max}$ are set to $I_{out} = 255$. Selection of the next point to move to is based on the roulette wheel technique. The selection probability (P) for each neighbour point (i.e. surrounding points in the search space) of ant is obtained as:

$$P = \frac{(1+\tau_i)^{\alpha} \times \left(\left(1+\left(\frac{k_i}{\gamma}\right)^{10}\right) \times \eta_i\right)^{\beta}}{\sum_{i \in G(i)} (1+\tau_i)^{\alpha} \times \left(\left(1+\left(\frac{k_i}{\gamma}\right)^{10}\right) \times \eta_i\right)^{\beta}} \tag{1}$$

where the parameters α and β control the relative importance of pheromone trail against heuristic value. $G(i)$ is the set of neighbourhood points around ant. τ_i is the amount of pheromone that exists in a neighbour point. In order to avoid the zero probability in the areas with no pheromone, the value 1 is added to τ_i. η_i is a heuristic value which for the neighbours of ant is set according to Fig. 4.

The heuristic values of neighbourhood are set aiming to have monotonically increasing transfer function. Also in (1), for up and right neighbours, k_i is set to $I_{in} - I_{in\text{-}min}$ in the horizontal axis and I_{out} in the vertical axis for the current position of ant, respectively and for the other neighbours it is set to zero. Thus, for right neighbour of ant, the value of k_i shows the distance the ant travels to up direction. Also for up neighbour, k_i shows the distance travelled by ant to the right. By use of k_i values, the parameter γ helps ants move to the target as:

$$\text{if } \frac{k_i}{\gamma} < 1 \rightarrow \left(\frac{k_i}{\gamma}\right)^{10} < \frac{k_i}{\gamma}, \quad \text{if } \frac{k_i}{\gamma} > 1 \rightarrow \left(\frac{k_i}{\gamma}\right)^{10} > \frac{k_i}{\gamma}. \tag{2}$$

The power exponent 10 provides enough strength for the technique mentioned in (2). Considering the ratio of $\frac{k_i}{\gamma}$ and the values of k_i, we can conclude that if an ant excessively moves up (i.e. increasing I_{out}), then the probability of moving to the right (with $k_i = I_{out}$) increases, that is:

$$\text{if } (I_{out}\uparrow) \,\&\, (I_{out} < \gamma) \rightarrow P_{right}\uparrow, \tag{3a}$$

$$\text{if } (I_{out}\uparrow) \,\&\, (I_{out} > \gamma) \rightarrow P_{right}\uparrow\uparrow. \tag{3b}$$

The above explanation holds true for the probability of up direction if ant moves to the right. Thus, this technique exponentially reminds the ants to go to the last point (i.e. $I_{in} = I_{in\text{-}max}$, $I_{out} = 255$). The value of γ defines the strength and the shape of force on ants. However, if ants continue their way and reach to the border of map (that is, $I_{in} = I_{in\text{-}max}$ or $I_{out} = 255$), they will be guided toward the last point automatically by the force of algorithm.

The parameters α, β, γ, C_{up}, and C_{right} are set by the genetic algorithm. Also, note that the value of γ which comes from GA, is in the range 0–255 for vertical direction (upside movement of ants) and it needs to be adjusted automatically in the range $I_{in\text{-}max} - I_{in\text{-}min}$ for horizontal direction (right side movement of ants).

We found that a population of 20 ants shows good behaviour. After all ants moved from the starting point to the last point, SA modifies the recent pheromone trail and transfer functions of ants. Then, the global pheromone update is performed as follows:

$$\tau_{ij}(t+1) = (1-\rho) \times \tau_{ij}(t) + \sum_{l=1}^{20} \Delta\tau_{ij}^{l}(t) \tag{4}$$

where ρ is the evaporation rate set to 0.4. $\Delta\tau_{ij}^{l}$ is the amount of pheromone deposited between the points i and j by the lth ant (which can be changed by SA) and it is equal to $\frac{F}{30 \times BF}$. F is the fitness function value of the lth ant (transfer function can be changed by SA). BF is the best found fitness value. It is used for normalisation of pheromone deposition, since fitness values may have various amounts for different image sizes. Also, the value of 30 in the denominator is to normalise τ_i with respect to the other parameters in (1).

The proposed ACO has elitism attribute (since pheromone is deposited in the trace of the best found transfer function). Deposited pheromone by the best found solution is proportional to evaporation rate as $\frac{\rho}{4} \times \Delta\tau_{ij}^{BF} = \frac{\rho}{4} \times \frac{BF}{30 \times BF} = \frac{\rho}{4} \times \frac{1}{30}$.

2.3. Genetic algorithm part

Genetic algorithm resembles Darwinian Theory of natural selection. GA initially produces a population of artificial chromosomes. After evaluating individuals in the population, GA selects parents and then produces children through the recombination process (crossover and mutation). The new acquired offspring will replace worse individuals in the population. In general this procedure leads the population of individuals to promising areas of search space.

A modified steady state genetic algorithm is used to facilitate faster convergence of ACO. The fundamental parts of GA are encoding, selection, and reproduction. The parameters of ACO (i.e. α, β, γ, C_{up}, and C_{right}) are encoded as a real coded chromosome with five genes. We consider 20 ants and 10 chromosomes, where each chromosome is responsible for adjusting the parameters of two ants. At the end of each generation, the individuals of population copy their values to the parameters of the corresponding two ants. The ranges of α and β are from 0 to 5, γ changes between 100 and 250, and those of C_{up} and C_{right} are set in the interval 0 to 3.

Parent selection is based on the roulette wheel technique. After all individuals were evaluated by the GA fitness function, two of them will be selected to produce two offspring. The worst individual and the weaker parent of the current generation are replaced with the two new offspring, which along with other individuals, form the next generation. Removing the worst individual is in the way of exploitation and deletion of weaker parent guaranties exploration. If the worst individual and the weaker parent are identical, then the two worst individuals will be replaced with the two new children. The procedure guaranties survival of the fittest because the best individual will not be deleted. The fitness function for GA is defined as:

$$F^{GA} = F_{mean}^{ant1} + F_{mean}^{ant2} + F_{best\,found}^{ant1,2} \tag{5}$$

where F^{GA} is the estimated fitness for each individual. F_{mean}^{ant1} and F_{mean}^{ant2} are the average fitness of transfer functions created by the two corresponding ants in ACO's iterations which are run between the two GA generations. $F_{best\,found}^{ant1,2}$ is the best found fitness value in the mentioned procedure.

Reproduction stage is carried out by crossover and mutation operators. Uniform crossover is used with the probability of 0.85.

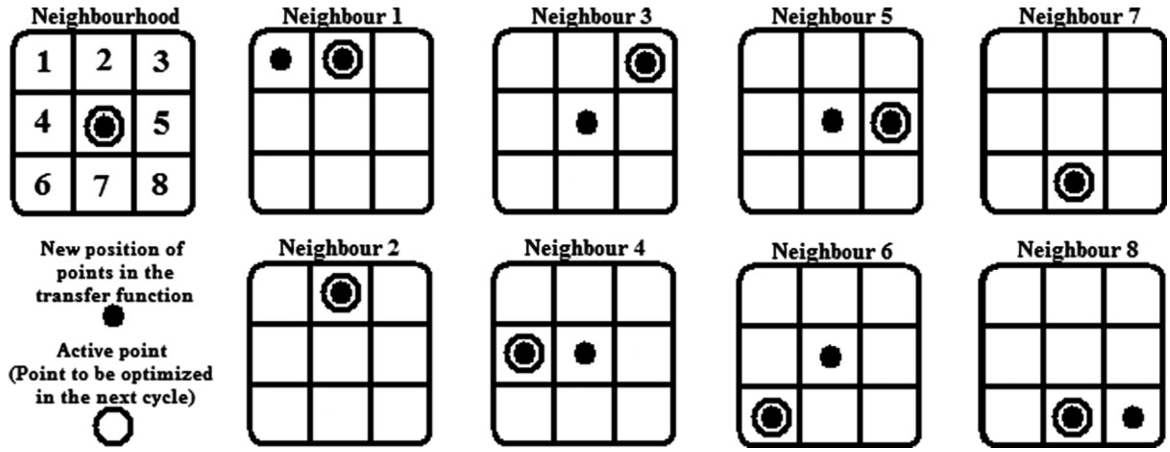


Fig. 5. Neighbourhood of each point in SA and the resulting action of choosing each of them.

Each offspring has mutation probability equal to 0.05. In our algorithm, mutation operator just changes one gene of chromosomes with the restriction of −10% to 10% of the original value of that gene. Note that the new values of genes cannot take quantities exceeding the range of corresponding parameters.

2.4. Simulated annealing part

Inspired from the annealing process of metallurgy, simulated annealing mimics controlled cooling of materials (slow and controlled cooling makes materials with less defects). An initial high temperature causes algorithm to have the chance of selecting bad solutions in a solution neighbourhood. By cooling in every step, the chance of choosing bad solutions is decreased. Hence in the last stages of simulated annealing, it converges and only selects good solutions.

The proposed hybrid algorithm uses simulated annealing in the local search phase of ACO. Almost optimum solutions are obtained after applying SA at the cost of increase in processing time. The essential components of SA are neighbourhood definition, probability function, initial temperature, and cooling schedule. In our work, SA optimises the best found transfer function and some randomly selected transfer functions obtained in the last run of ACO. Modification of the best found transfer function by SA has a great chance to find a better solution. We use random points as the starting points of SA's optimisation process in the selected transfer functions. Also ant's pheromone trail corresponding to its optimised transfer function is adjusted according to the changes made in the transfer function.

Fig. 5 depicts the neighbourhood of each point in the transfer function and the resulting action in case of choosing one of the neighbours. Hollow circle shows the active point (point to be optimised in the next cycle) and the filled circle demonstrates the modified location of current selected point or its adjacent points. As shown in Fig. 5, by selecting the neighbour no. 3 or no. 5, the modification in the next cycle will be performed on the point in the right side of current point. It holds true for the point in the left side of current point when one of the neighbours no. 4 or no. 6 is selected. Among the neighbours of each point, only selection of the neighbour no. 2 or no. 7 has no effect on the adjacent points of the current one. Note that, only the neighbours are allowed in which, monotonically increasing form of transfer functions is preserved. Further, if at last none of the neighbours of the current point are selected (regarding the probability function), then the optimisation of that point will be terminated (i.e. that point is a local optimum).

The probability function for selecting a neighbour based on the analogy with physical systems is defined as:

$$\begin{cases} P = e^{\frac{F^{new} - F^{old}}{0.05 \times F^{old} \times T}} & \text{if } F^{new} < F^{old}, \\ P = 1 & \text{if } F^{new} > F^{old}, \end{cases} \tag{6}$$

where F^{old} is the current fitness value, F^{new} is the resulted fitness value, and T is the temperature. F^{old} is used at the denominator of the exponent to normalise the numerator, since the numerator has various quantities for different image sizes. The initial temperature was set to 200. The cooling schedule for the proposed hybrid model is defined as:

$$T(t+1) = T(t) \times 0.5. \tag{7}$$

2.5. Fitness criterion

Fitness function is the bottleneck of the algorithm from the point views of quality of the obtained images and computational cost. Automatic image enhancement needs a fitness function to be independent from any parameter which should be set by human (such as the parameters used in [11]). On the other hand, only multiplication of operands (as used in [12,13]) is not enough. The reason is that if a processed image has lower values for all operands in contrast to the original image, then the multiplication of operands in the fitness function shows extra low value for weaker image. In order to overcome this problem, we define the fitness function as the geometric mean of the operands, that is:

$$F = \sqrt[3]{STD \times ENTROPY \times SOBEL} \tag{8}$$

where *STD* is the global standard deviation of intensities (contrast measure), *ENTROPY* is the global entropy (randomness measure) of grey levels in image, and *SOBEL* is defined as [17]:

$$SOBEL = mean\left(|sobel_{vertical}| + |sobel_{horizontal}|\right) \tag{9}$$

where $sobel_{vertical}$ and $sobel_{horizontal}$ are the images obtained by applying the vertical and horizontal Sobel operators respectively, and *mean*(.) operator denotes averaging. Sobel as a gradient operator is frequently used for edge detection of images. Global standard deviation of an image is a measure of the dispersion of pixels' intensity from their mean. The more spread apart the intensity levels, the higher the deviation. Standard deviation is calculated as the square root of variance. Entropy is a measure of disorder, i.e. the more disorder the more entropy. Global entropy of an image is calculated according to the following equation [17]:

$$Entropy = -\sum_i \sum_j c_{ij} \times \log_2 c_{ij} \tag{10}$$

where c_{ij} is the intensity of a pixel across the image. The span of i and j is limited to image dimensions.

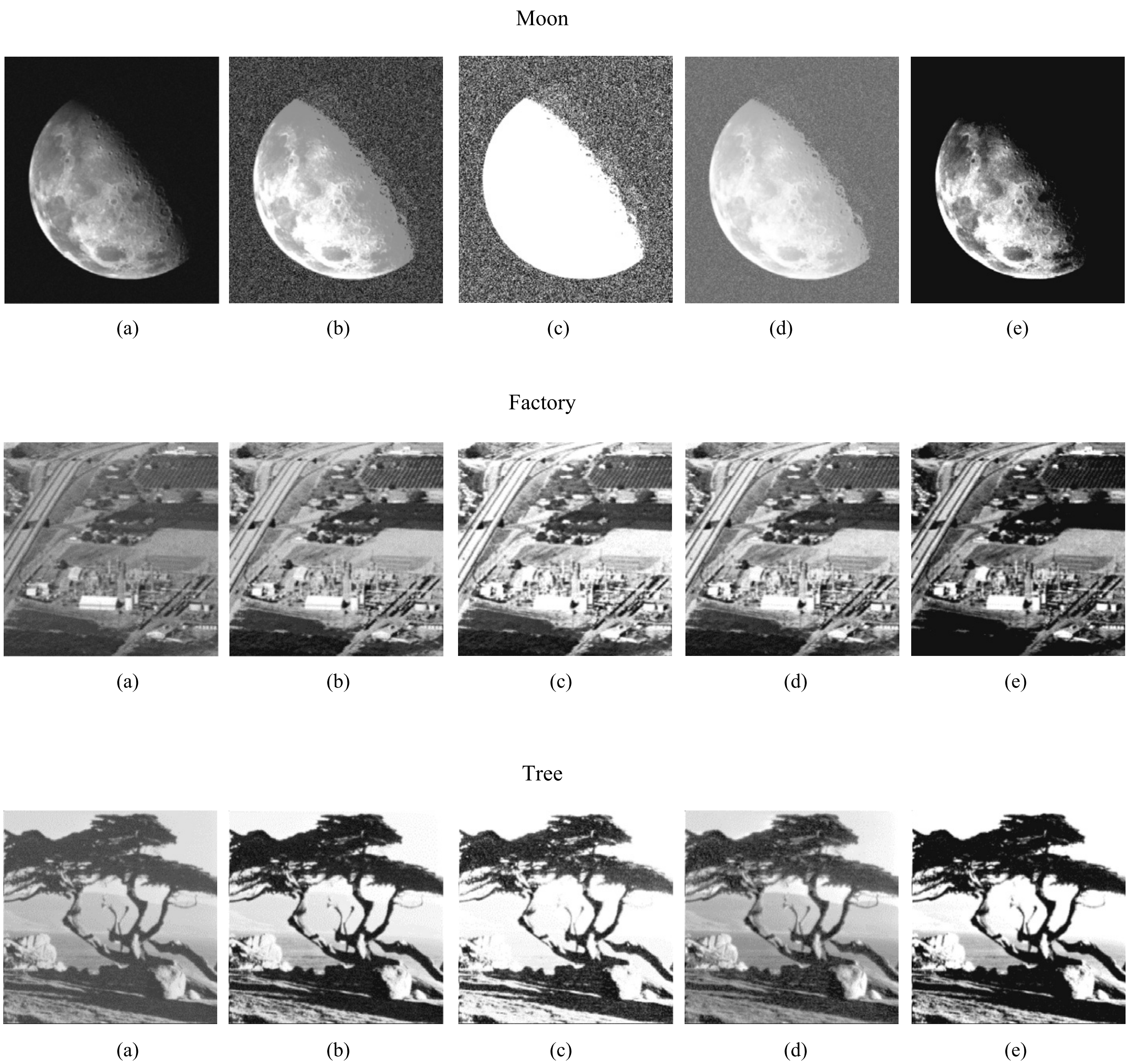


Fig. 6. Results of applying the proposed method with and without the entropy operator: (a) original, (b) proposed method, (c) proposed method without entropy operator, (d) histogram equalisation, (e) fuzzy method.

Local operators are good candidates for the fitness criterion. Nevertheless, global standard deviation and entropy are used instead of local operators to facilitate less computation time of fitness function. Local operators work on each pixel and their neighbourhood, whereas global operators perform one time processing on all the pixels of an image. Standard deviation is a measure of contrast in images but it is not able to correctly determine the enhanced images alone. Therefore, the Sobel operator is also used to determine contrast of images. Since images with high contrast have more detectable edges compared to low contrast ones, so Sobel operator as a reliable and fast edge detector is a good criterion.

The motivation of utilising global standard deviation along with Sobel edge detector is that the Sobel edge detector alone may result in images that have more edges but do not have global contrast in other areas. Even though Sobel as a local operator is very effective, a global operator is required to ensure that the contrast enhancement will be covered within all areas of images not just at the edges. Therefore global standard deviation as a global contrast measure is combined with Sobel edge detector.

Entropy for preventing binary outputs and noise intensification Although STD and Sobel operators tend to create high contrast images, sometimes they may result in artificial images with low diversity in intensities. Hence, entropy which measures the randomness in images has been used in (8) to make sure that the obtained images will have variety of intensity ranges and look natural.

Experimental results show the efficiency of using the standard deviation and Sobel edge detector, but without the entropy operator in some cases the algorithm may result in a black and white image. Fig. 6 shows what happens in the absence of entropy

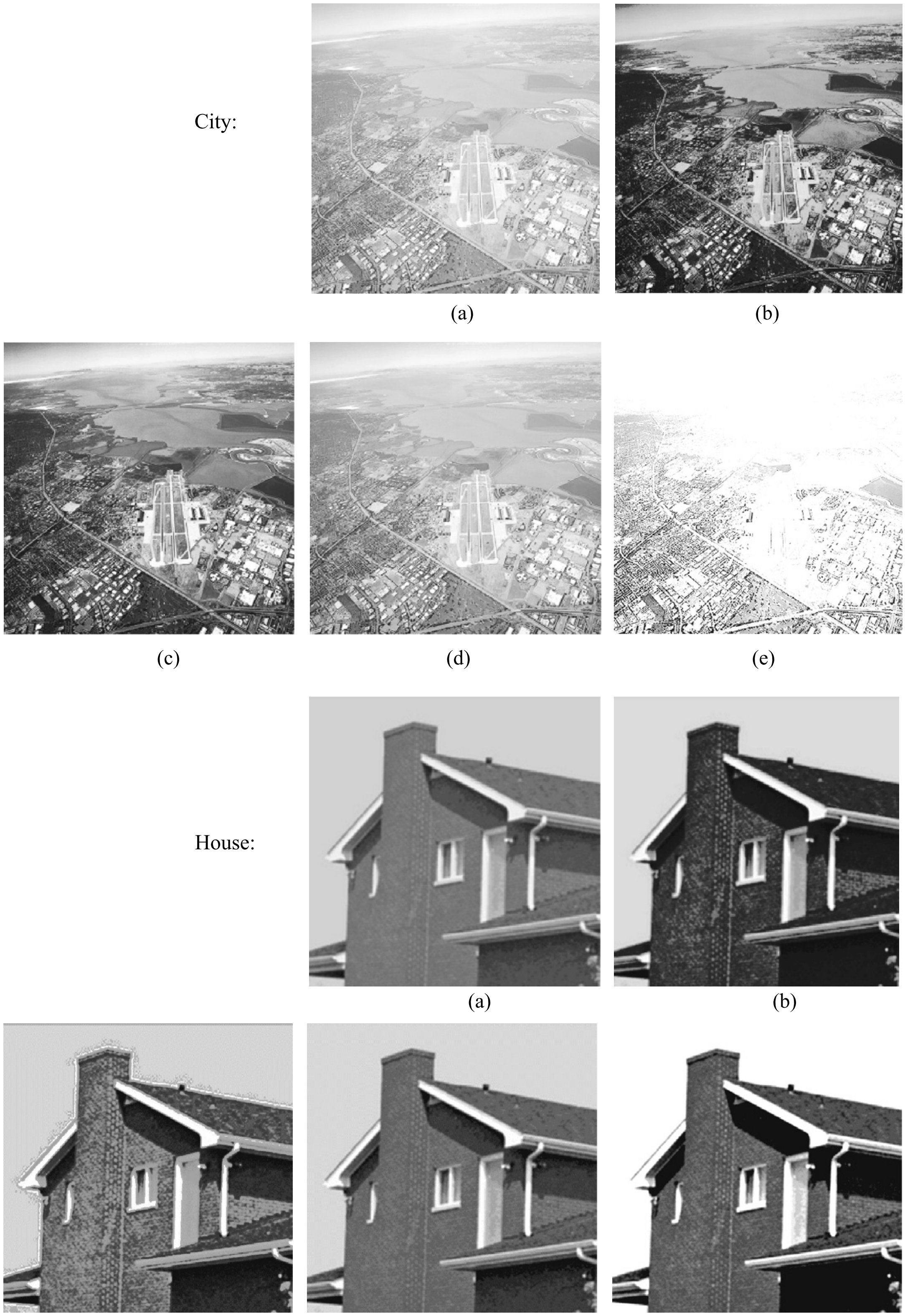
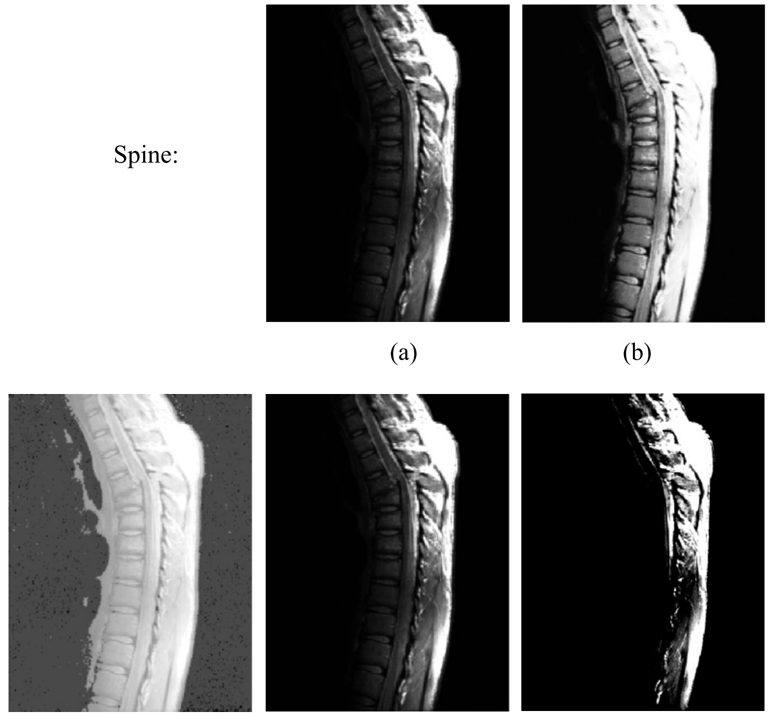


Fig. 7. Results of different enhancement techniques: (a) original, (b) proposed method, (c) histogram equalisation, (d) linear contrast stretching, (e) fuzzy method.

for some test images. Although most contrast increasing methods are sensitive to noise, absence of the entropy in the presented method deteriorates the results obtained from a noisy image. Fig. 6 (Moon) illustrates the results obtained from applying the proposed method with and without entropy operator along with histogram equalisation results for an image with Gaussian noise. The additive Gaussian noise has zero mean and its variance is 0.001 of total intensity range. In the displayed results

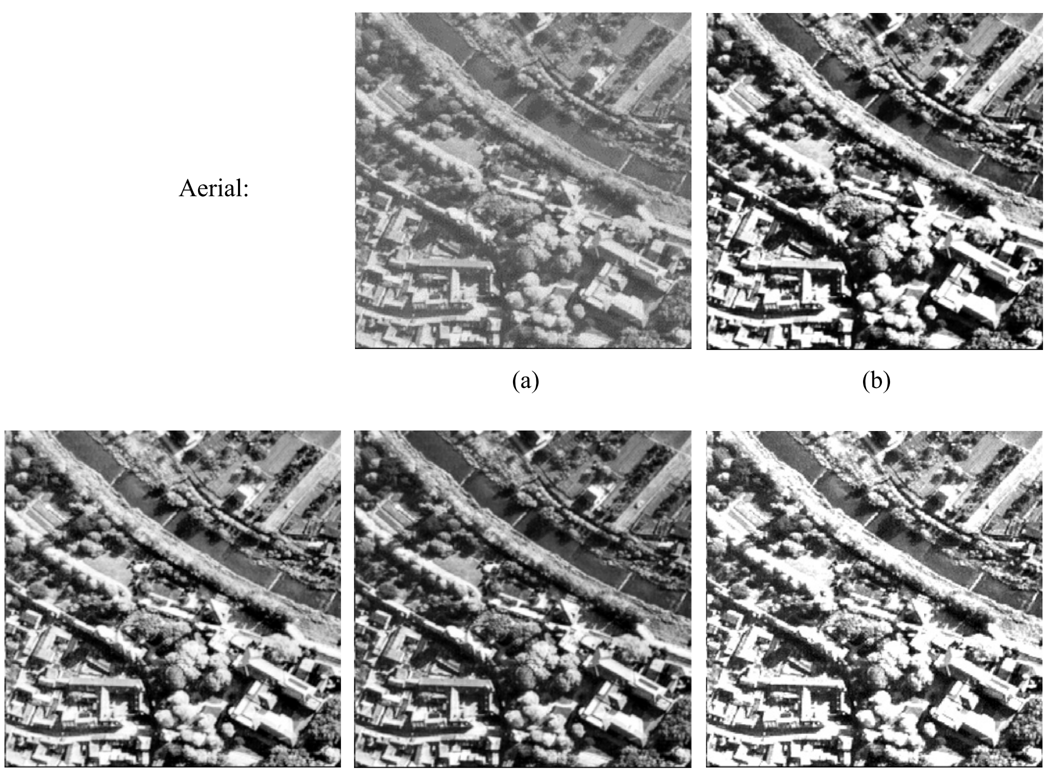
Spine:



(a) (b)

(c) (d) (e)

Aerial:



(a) (b)

(c) (d) (e)

Fig. 7. (*continued*)

it is obvious that absence of the entropy has led to worse results.

Eqs. (8) and (9) indicate that the proposed algorithm searches for large global standard deviation values (in the sense of high global contrast) and large values of entropy. Entropy factor guaranties that images seem more natural, since it prevents image from binarisation. This method also attempts to increase the edges of images (as high contrast images have more edges).

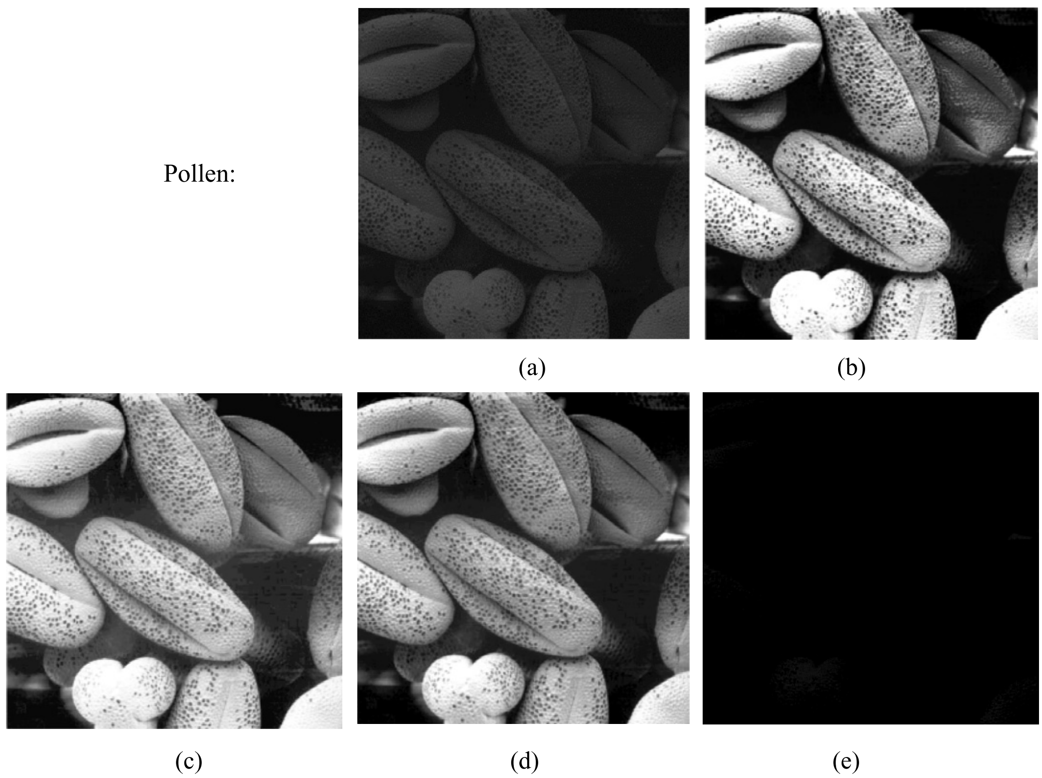


Fig. 7. (*continued*)

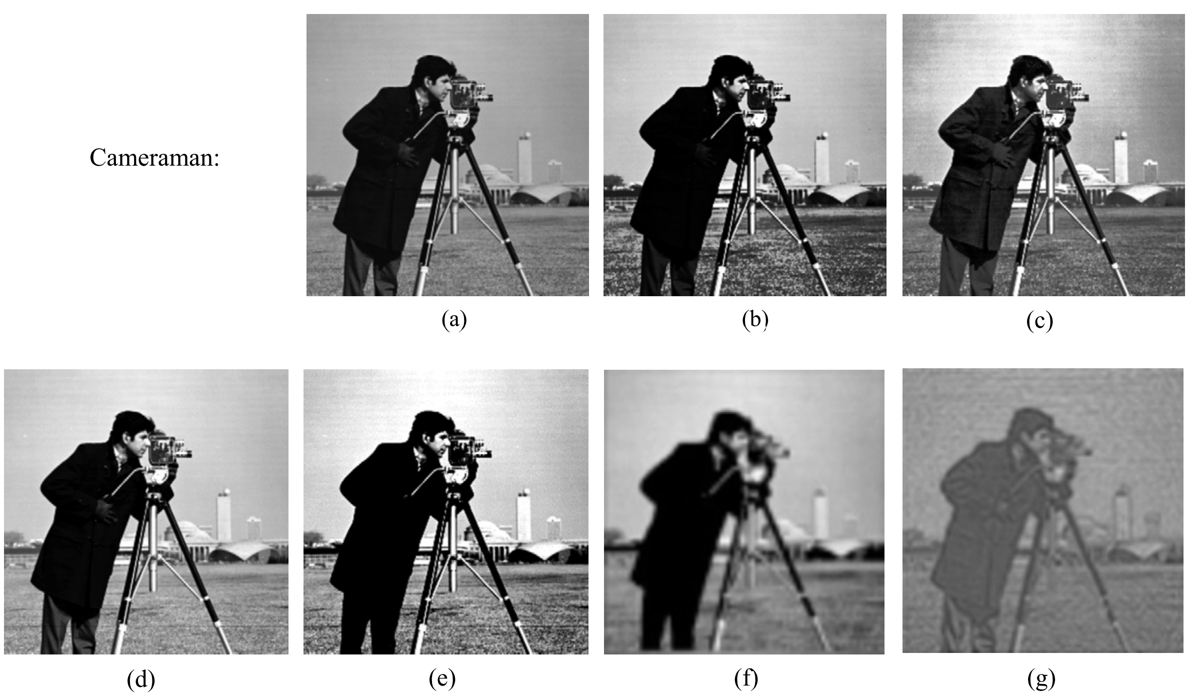


Fig. 8. Comparison of different enhancement methods: (a) original, (b) proposed method, (c) histogram equalisation, (d) linear contrast stretching, (e) fuzzy method, (f) method of [12], (g) method of [13]. Note: images from references [12] and [13] are at lower resolution.

3. Experimental results

In order to evaluate the efficiency of the proposed algorithm, we apply it to the several common test images used in the literature. The iteration number is set to 100. We also apply several other enhancement methods to compare their performance with the proposed method.

Fig. 7 illustrates the results of applying the new method, linear contrast stretching [17], histogram equalisation [17], and fuzzy technique (with the rules described in [17]) to the five test images. We observe that the proposed method has better performance. Furthermore, Fig. 8 shows the efficiency of our technique compared to the above mentioned methods and also the methods of [12] and [13]. Fig. 9 depicts the transfer functions obtained by

Pout:

Tire:

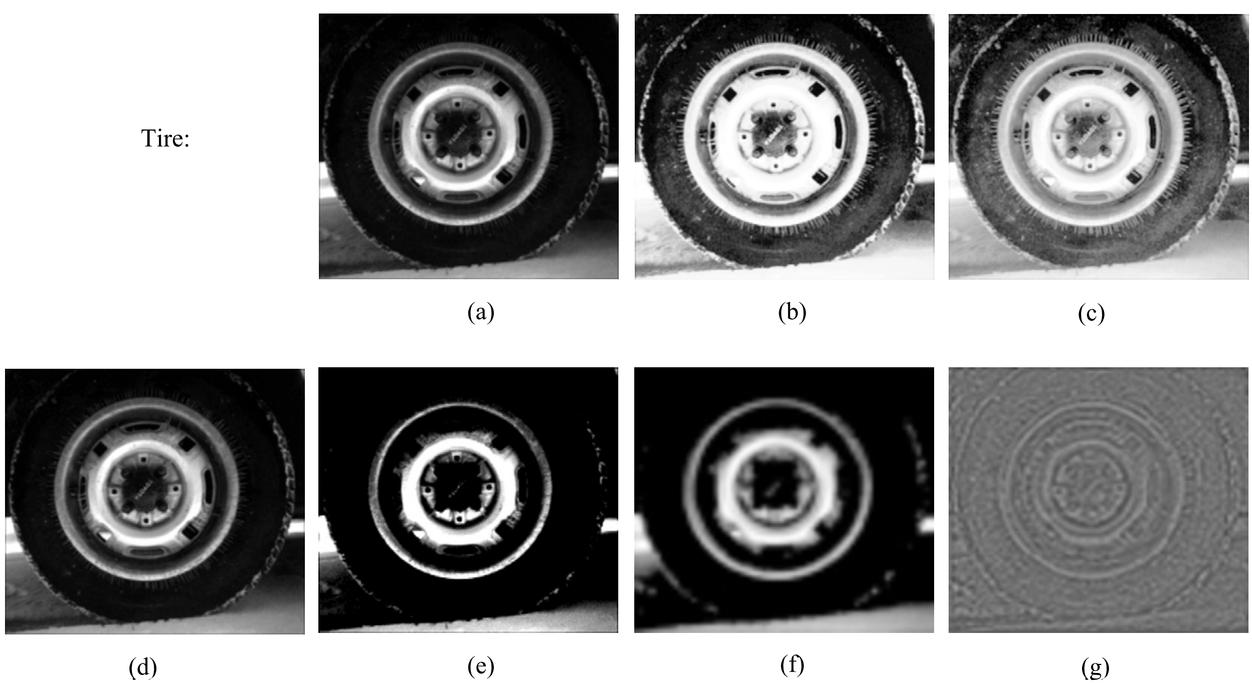


Fig. 8. (*continued*)

applying of the proposed method to the test images. Also, Fig. 10 demonstrates the final pheromone trail of ants in search panel. It is observed that ACO converges in all cases.

In order to verify the proposed technique from objective viewpoint, Tables 2 and 3 present the values of two statistical criteria obtained from different methods. Uniformity and contrast are two descriptors of co-occurrence matrix [17]. High values of contrast and low values of uniformity are desired characteristics. It is observed that the proposed method satisfies the above two criteria well.

Table 4 shows the fitness function values defined in (8) for different methods. It is observed that the proposed algorithm yields higher fitness values.

Detail variance (DV) and background variance (BV) are the other objective measurements [18]. It is desirable that the enhancement methods result in image with higher DV compared to the original image while the BV does not change. Higher DV value represents higher contrast in image. Great increase in the BV value of image after performing an enhancement method indicates that the method is sensitive to the original image noise. Tables 5 and 6

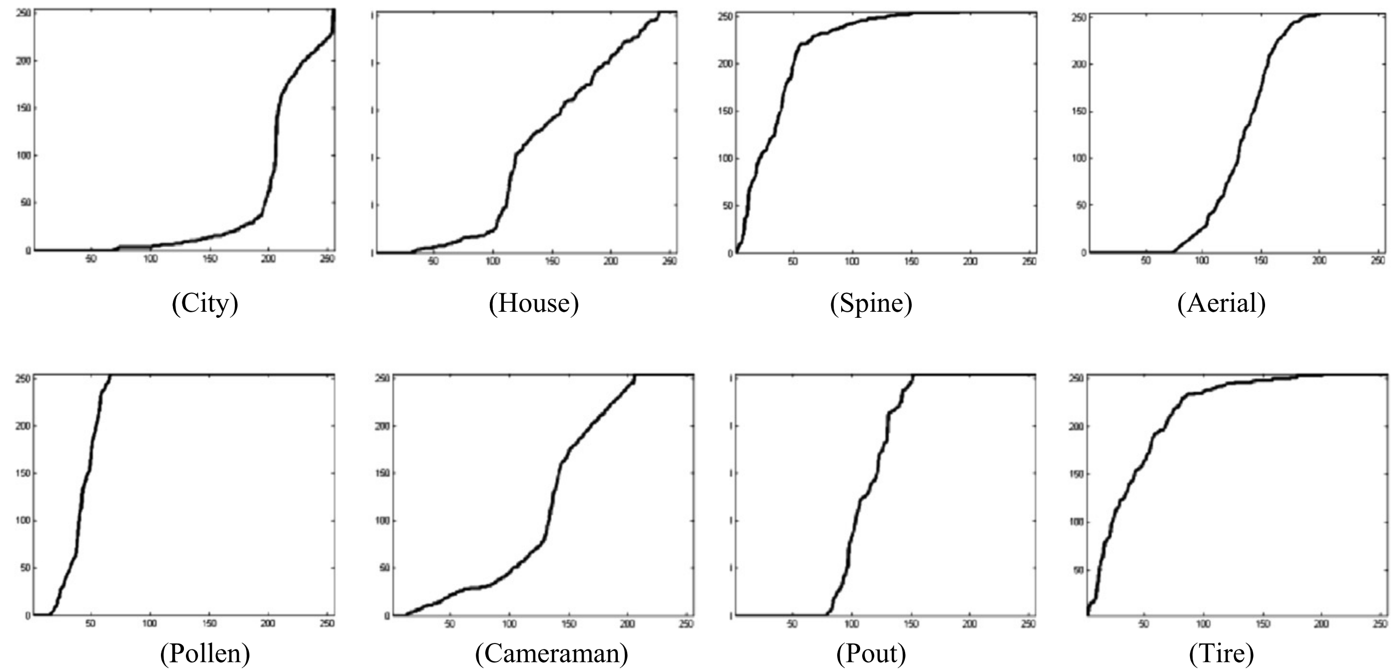


Fig. 9. Transfer functions obtained by the proposed method for test images (horizontal axis and vertical axis are the input and output intensities, respectively).

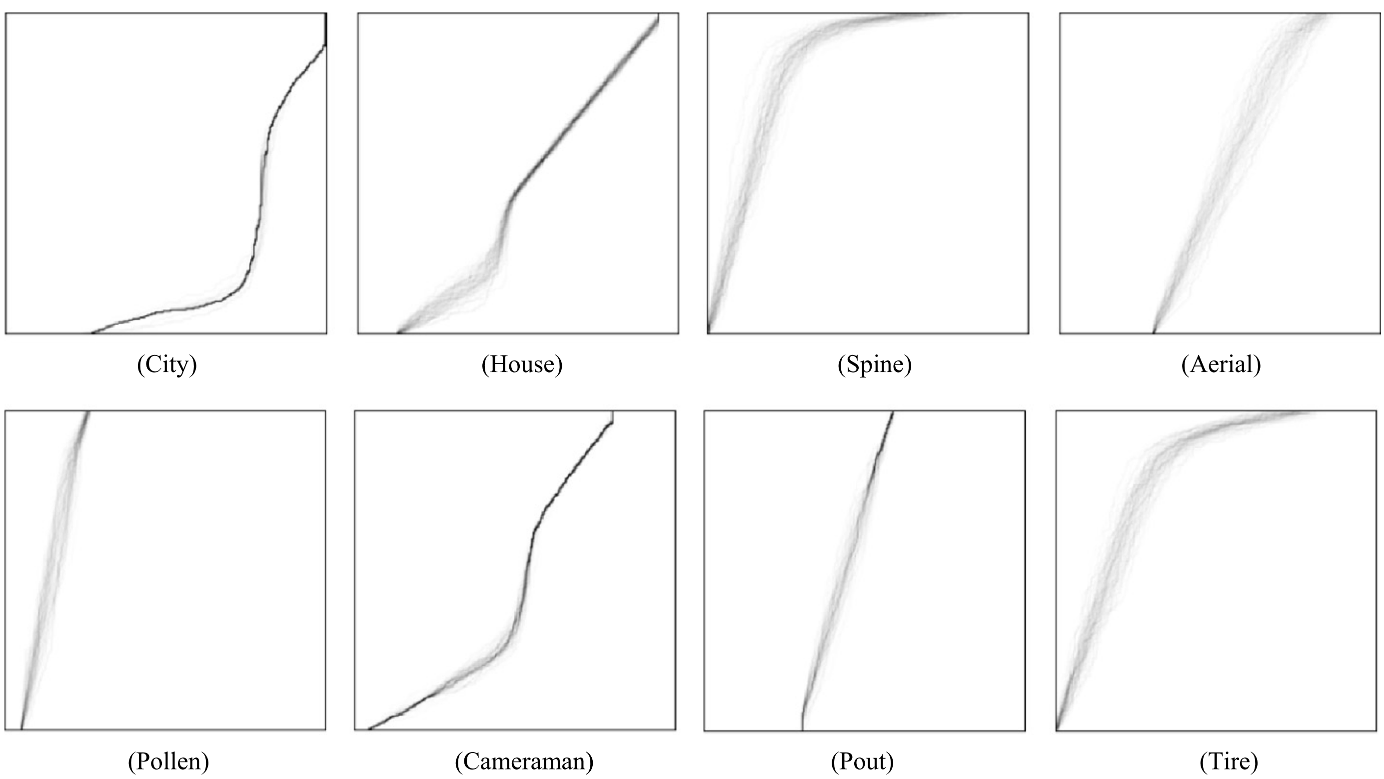


Fig. 10. Final pheromone trail of ants in search space (lower bound is the input intensity axis and left bound is output intensity axis).

demonstrate the DV and BV values of all test images respectively for different methods where a 3×3 local STD mask and threshold value of 30 are used. We again observe that the new method satisfies the above criteria more than the other methods.

Repeatability of the proposed algorithm can be proved by Fig. 11 which shows the obtained transfer functions for each of ten times run of the algorithm for each test image. As illustrated, the resulted transfer functions almost have the same shapes in different runs. However, in some cases we observe that more than one optimum exist on the transfer function pane, for example in the *cameraman* image. Thus, another measurement is needed to evaluate the robustness of the new method.

The fitness value augmentation process in 100 iterations of the algorithm for different test images is shown in Fig. 12. It is observed that in all cases the algorithm is terminated at the 100th iteration which is due to the fixed stop condition set to 100. Predefined stopping condition assures that the algorithm finishes its process in an expected period. Fig. 12 also shows the robustness

Table 2
Uniformity descriptor for experimental images.

Image	Original	Proposed method	Histogram equalisation	Linear contrast stretching	Fuzzy
City	0.20	0.10	0.05	0.13	0.53
House	0.26	0.21	0.13	0.18	0.23
Spine	0.50	0.40	0.34	0.50	0.73
Aerial	0.11	0.05	0.04	0.06	0.07
Pollen	0.44	0.10	0.08	0.08	0.99
Cameraman	0.16	0.13	0.07	0.14	0.13
Pout	0.28	0.08	0.07	0.10	0.16
Tire	0.23	0.09	0.06	0.22	0.50

Table 3
Contrast descriptor for experimental images.

Image	Original	Proposed method	Histogram equalisation	Linear contrast stretching	Fuzzy
City	0.55	1.55	1.18	0.86	1.41
House	0.05	0.12	0.18	0.10	0.12
Spine	0.08	0.16	0.10	0.15	0.08
Aerial	0.43	1.38	0.98	0.90	1.27
Pollen	0.06	0.46	0.28	0.26	0.00
Cameraman	0.50	1.04	0.73	0.74	1.16
Pout	0.08	0.33	0.29	0.26	0.19
Tire	0.23	0.58	0.47	0.23	0.34

Table 4
Fitness values of experimental images.

Image	Original	Proposed method	Histogram equalisation	Linear contrast stretching	Fuzzy
City	25.40	34.38	33.50	29.11	18.84
House	19.17	24.57	24.11	21.45	23.31
Spine	14.50	17.32	14.84	14.50	9.65
Aerial	29.40	39.83	36.23	37.48	37.53
Pollen	11.22	30.97	28.22	27.84	0.83
Cameraman	24.95	29.77	27.69	28.49	29.32
Pout	13.39	28.89	27.72	26.19	22.39
Tire	25.89	31.63	28.89	26.12	18.81

of simulated annealing which helps further fitness growth in the last stages of the algorithm. Thus the proposed method operates in a limited time interval and performs optimisations up to final iterations. Regarding the simulated annealing functionality in hybrid algorithm, we can conclude that the fitness convergence is not intended to be obtained before finishing the algorithm. In fact the output image after termination of the algorithm is the result of continuous fitness growth during the fixed optimisation cycle.

Table 7 illustrates the average and standard deviation of DV, BV, and fitness values of ten times run of the algorithm. It is observed that the STD values are low in comparison with the corresponding average values which mean different runs achieve almost the same response.

It should be noted that the proposed method takes rather more processing time compared to the classical methods. This is expected since metaheuristics generally perform search loops. In each loop, fitness evaluation is the most time consuming operation. Global transformation has advantage of being faster than the local processing of images [12,13]. On the other hand, simulated annealing is used in our method to emphasis the quality, although it increases computational cost of the algorithm. Processing time is an important matter in some applications so it can be lessened by decreasing the total iteration numbers of algorithm or by reducing the use of SA in the algorithm (i.e. reducing SA's executions, the number of transfer functions and points of each transfer function to be optimised by SA, and the number of cycles of SA optimisation process). Table 8 shows the effect of different iteration numbers on the run time and the best found fitness values for two test images in two cases. In the first case simulated annealing is used as the local search and in the second case it was disabled. As expected lower iteration number or disabling the local search (SA) results in less elapsed time and less fitness value.

As mentioned, classical methods have the advantage of less computational time. However, for fair comparison of the proposed method with competitive algorithms (like [11–13]), it is necessary that parameters such as time spent, number of iterations, the system on which the test is performed be given and also the test images be identical. Since all information required for comparison of the proposed algorithm with similar probabilistic methods has not been provided, we presented the computation time of our tests and their respective specifications in Table 9. Respecting machine computation power and programming language which are common and average in speed, the time spent for small images such as Tire (205×232 pixels) seems reasonable (37 s) whereas for large images like Spine (976×746 pixels) it may be tolerable (265 s). An important point is that by increasing image size, processing time does not grow linearly with size. For example, the picture Spine is 15.27 times larger than the picture Tire but its computation time is just $265/37 = 7.16$ times longer. Nonetheless it is obvious that the proposed method is only suitable for offline applications.

Note that the proposed method is able to enhance the contrast of colour images simply by performing it on the intensity component of HSI (hue, saturation, intensity) colour model of images [17]. HSI colour space can be obtained from RGB colour system.

Table 5
Detail variance (DV) of the experimental images using a 3×3 local STD mask. Threshold limit of DV is 30. NaN means there is no pixel with local STD greater than threshold number.

Image	City	House	Spine	Aerial	Pollen	Cameraman	Pout	Tire
Original	47.30	34.95	44.84	39.63	NaN	53.42	34.00	41.45
Proposed method	63.31	41.81	46.81	54.42	44.75	56.80	42.38	48.26
Histogram equalisation	56.28	40.64	35.63	48.04	39.00	60.38	41.18	43.27
Linear contrast stretching	54.22	36.65	44.84	47.70	38.43	61.35	44.59	41.60
Fuzzy	72.39	45.95	53.58	53.27	NaN	60.30	43.68	53.53

Table 6
Background variance (BV) of the experimental images using a 3×3 local STD mask. Threshold limit of BV is 30.

Image	City	House	Spine	Aerial	Pollen	Cameraman	Pout	Tire
Original	8.47	3.48	2.05	13.44	2.87	5.61	3.18	7.44
Proposed method	8.19	4.95	2.92	13.92	9.66	5.32	9.47	8.93
Histogram equalisation	11.57	6.59	1.73	15.93	10.17	8.77	9.77	10.86
Linear contrast stretching	9.12	3.90	2.05	15.21	10.16	6.60	8.25	7.50
Fuzzy	3.56	4.07	1.11	13.44	0.21	5.87	6.60	3.25

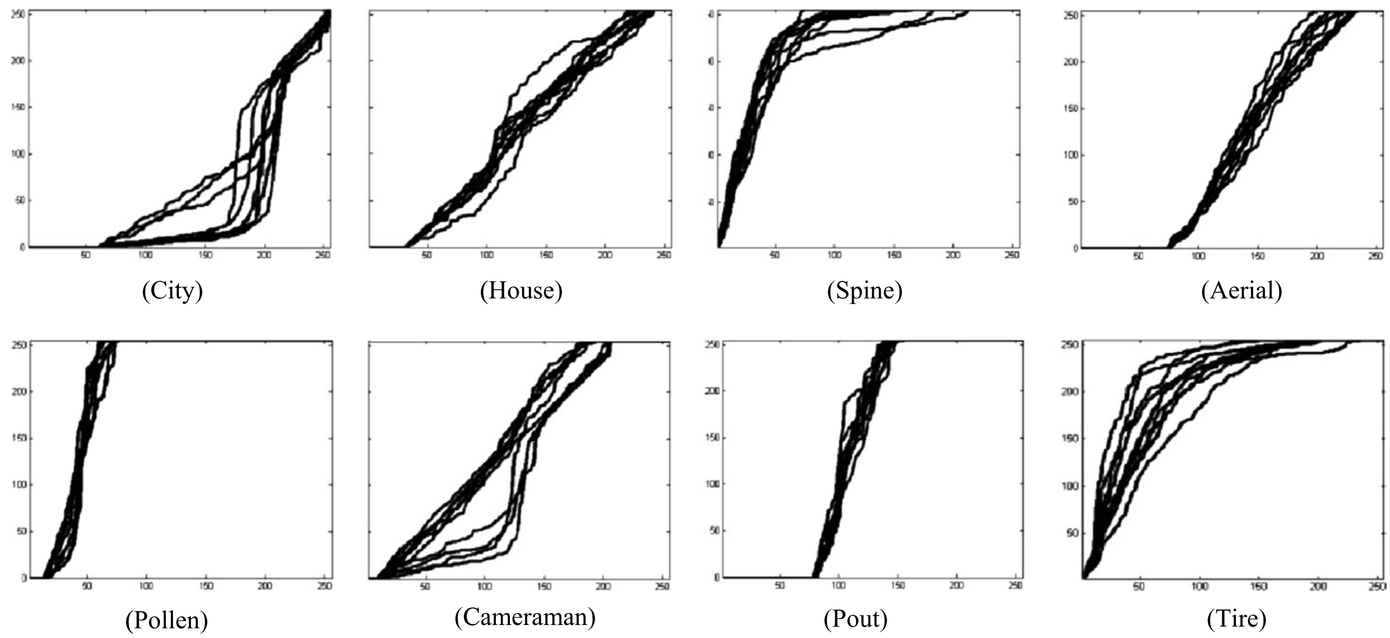


Fig. 11. Transfer functions obtained from each of ten times run of the proposed algorithm for test images (horizontal axis and vertical axis are input and output intensities, respectively).

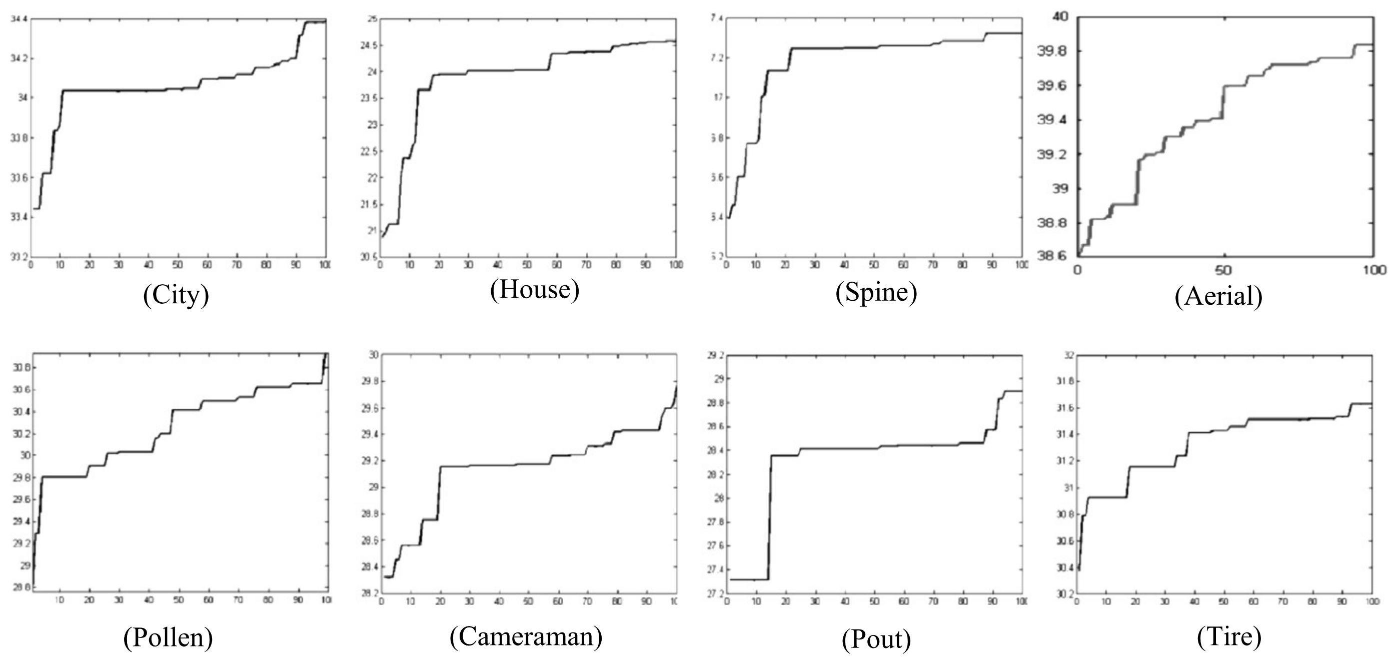


Fig. 12. Fitness evolution of test images during the proposed algorithm progress (horizontal axis and vertical axis are iterations of algorithm and the best found fitness, respectively).

Table 7
Average and standard deviation of DV, BV, and fitness of ten times run of the proposed algorithm.

Image	Fitness		DV		BV	
	Average	STD	Average	STD	Average	STD
City	33.91	1.15	62.64	7.32	8.79	1.63
House	16.66	0.21	38.83	1.65	2.47	0.28
Spine	17.03	0.27	43.59	1.29	2.54	0.07
Aerial	38.82	0.59	52.69	1.93	14.15	0.66
Pollen	30.49	0.56	45.97	3.49	8.59	1.01
Cameraman	29.90	0.29	59.47	1.72	5.90	1.26
Pout	29.34	0.27	43.14	1.90	9.24	0.48
Tire	31.64	0.32	50.61	2.73	8.26	0.58

Table 8
Effect of iteration number and local search on the best found fitness and run time.

Image	Iteration number	Simulated annealing used	Fitness	Elapsed time (seconds)
Aerial	150	Yes	39.87	64
	150	No	39.39	38
	100	Yes	39.83	43
	100	No	39.38	24
	80	Yes	39.47	34
	80	No	39.23	21
	60	Yes	38.69	29
	60	No	37.28	16
Tire	150	Yes	31.64	56
	150	No	31.41	36
	100	Yes	31.63	37
	100	No	31.06	22
	80	Yes	31.55	26
	80	No	30.17	19
	60	Yes	30.90	21
	60	No	29.05	15

Table 9
Processing time for each test image.

Image	Size	Programming language	Machine	Algorithm iterations	Duration (seconds)
City	769×765	Matlab	2×2.4 GHz	100	219
House	512×512	Matlab	2×2.4 GHz	100	110
Spine	976×746	Matlab	2×2.4 GHz	100	265
Aerial	256×256	Matlab	2×2.4 GHz	100	43
Pollen	500×501	Matlab	2×2.4 GHz	100	108
Cameraman	256×256	Matlab	2×2.4 GHz	100	49
Pout	291×240	Matlab	2×2.4 GHz	100	47
Tire	205×232	Matlab	2×2.4 GHz	100	37

Table 10
Difference of the proposed method with analogous works.

Properties of method	Proposed	[10]	[11]	[12]	[13]
Global or local contrast enhancement	Global (fast)	Global (fast)	Global (fast)	Local (slow)	Local (slow)
Evolutionary algorithm	ACO, SA, GA (slow, robust)	GA (fast)	GA (fast)	GA (fast)	Particle swarm optimisation (fast)
Automatic fitness criterion	Yes	Needs an operator for subjective fitness evaluation	Needs a coefficient to be specified	Yes	Yes

Briefly there is no difference between colour images and grey level images in enhancing their contrasts using the proposed method. For example, the images Tree and House are the intensity components of their corresponding colour images.

3.1. *Summary of comparisons*

In this subsection, a summary of comparisons made in previous sections with related works is briefly described. The methods in [10–13] are close to our work because of using evolutionary algorithms for image contrast enhancement. We categorise the properties of each method into two groups, namely evolutionary scheme used, and fitness criterion. In Table 10 the difference and similarities of the proposed method with analogous works are presented based on the aforementioned categories of comparison.

4. Conclusions

In this paper, a new hybrid approach was presented to enhance the contrast of images. The proposed SA and ACO are novel approaches used in the global transformation of images. The presented GA helps in the automation of the new method. The fitness function used in this work is an improved criterion which results in good performance. It makes the processed images seem natural. We compared the new method with several commonly used techniques such as linear contrast stretching, histogram equalisation, fuzzy, and the methods of [12] and [13]. The results indicate that the new method outperforms the above mentioned techniques from the subjective and objective viewpoints. We also found that the proposed method achieves the desirable uniformity, contrast, DV, and BV values better than the other methods. The new method can be used to improve images from point view of human observer or as a pre-processing task for other image processing applications.

References

[1] T. White, B. Pagurek, F. Oppacher, ASGA: Improving the ant system by integration with genetic algorithms, in: Proceedings of the Third Annual Conference on Genetic Programming, 1998.

[2] Z.J. Lee, S.F. Su, C.C. Chuang, K.H. Liu, Genetic algorithm with ant colony optimization (GA–ACO) for multiple sequence alignment, Appl. Soft Comput. 8 (2008) 55–78.

[3] S. Nemati, M.E. Basiri, N. Ghasem-Aghaee, M.H. Aghdam, A novel ACO–GA hybrid algorithm for feature selection in protein function prediction, Expert Syst. Appl. 36 (2009) 12086–12094.

[4] A. Rezaee, Extracting edge of images with ant colony, J. Electr. Eng. 59 (1) (2008) 57–59.

[5] J. Tian, W. Yu, S. Xie, An ant colony optimization algorithm for image edge detection, in: IEEE Congress on Evolutionary Computation, Hong Kong, June 2008, pp. 751–756.

[6] J. Tian, L. Ma, W. Yu, Ant colony optimization algorithm for wavelet-based image interpolation using a three-component exponential mixture model, Expert Syst. Appl. 38 (10) (2011) 12514–12520.

[7] J. Tian, W. Yu, L. Ma, AntShrink: Ant colony optimization for image shrinkage, Pattern Recogn. Lett. 31 (13) (2010) 1751–1758.

[8] A.R. Malisia, H.R. Tizhoosh, Applying ant colony optimization to binary thresholding, in: IEEE ICIP, 2006, pp. 2409–2412.

[9] R. Moussa, M. Beurton-Aimar, P. Desbarats, On the use of social agents for image segmentation, in: International Conference on Complex Systems and Applications (ICCSA 2009), Le Havre, France, 2009.

[10] C. Munteanu, V. Lazarescu, Evolutionary contrast stretching and detail enhancement of satellite images, in: Proceedings of Mendel, Czech Republic, 1999, pp. 94–99.

[11] F. Saitoh, Image contrast enhancement using genetic algorithm, in: IEEE International Conference on Systems, Man, and Cybernetics, 1999.

[12] C. Munteanu, A. Rosa, Towards automatic image enhancement using genetic algorithms, in: Proceedings of the Congress on Evolutionary Computation, vol. 2, 2000.

[13] M. Braik, A. Sheta, A. Ayesh, Image enhancement using particle swarm optimization, in: Proceedings of the World Congress on Engineering (WCE), vol. 1, UK, July 2007.

[14] R. Poli, S. Cagnoni, Evolution of pseudo-colouring algorithms for image enhancement with interactive genetic programming, Technical Report: CSRP-97-5, Univ. of Birmingham, January 1997.

[15] D.D. Duc, H.Q. Dinh, H.H. Xuan, On the pheromone update rules of ant colony optimization approaches for the job shop scheduling problem, in: Intelligent Agents and Multi-Agent Systems, in: Lecture Notes in Computer Science, vol. 5357, Springer, 2008, pp. 153–160.

[16] H. Huang, C.G. Wu, Z.F. Hao, A pheromone-rate-based analysis on the convergence time of ACO algorithm, IEEE Trans. Syst. Man Cybern., Part B, Cybern. 39 (2009) 910–923.

[17] R.C. Gonzalez, R.E. Woods, Digital Image Processing, 3rd edition, Prentice Hall, 2008.

[18] G. Ramponi, A cubic unsharp masking technique for contrast enhancement, Signal Process. 67 (2) (1998) 211–222.

Pourya Hoseini received the B.S. degree in electrical and electronics engineering from Azad University of Lahijan, Lahijan, Iran in 2007, and the M.S. degree in electronics and microelectronics engineering from Urmia University, Urmia, Iran in 2011. His research interests are analog and digital circuit design, hardware implementation of artificial intelligence, fuzzy systems, neural networks, evolutionary algorithms, and image processing.

Mahrokh G. Shayesteh received the B.S. degree from University of Tehran, Tehran, Iran, the M.S. degree from Khajeh Nassir University of Technology, Tehran, Iran, and the Ph.D. degree from Amir Kabir University of Technology, Tehran, Iran, all in electrical engineering. She is now an associate professor at Urmia University. She is also working with the Wireless Research Lab., ACRI, Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran. Her research interests are wireless communication systems, spread spectrum, and image processing.