Amal AbuNaser, Iyad Abu Doush*, Nahed Mansour and Sawsan Alshattnawi

# Underwater Image Enhancement Using Particle Swarm Optimization

**Abstract:** This article introduces a framework for enhancing underwater images using the particle swarm optimization algorithm. A pre-processing step is introduced to reduce the absorbing and scattering effects of water before applying a filter based on this algorithm to enhance the image. The quality of enhanced images is quantitatively assessed by applying the framework on a dataset of underwater images. The obtained results show a considerable improvement.

**Keywords:** Underwater images, particle swarm optimization, underwater image enhancement, Kullback–Leibler divergence, histogram, peak signal-to-noise ratio, number of edges.

DOI 10.1515/jisys-2014-0012
Received February 6, 2014; previously published online August 7, 2014.

## 1 Introduction

Underwater images usually suffer from light absorption, back scattering, and forward scattering. Back scattering is a fraction of the light power reflected by the water back to the camera before it reaches the image. Forward scattering is a random light issued by the object toward the camera [2]. Another problem that decreases the quality of underwater images is floating particles in water [1].

The level of noise introduced by these effects depends on sea structure, water quality, etc. [20]. As the distance increases when shooting the underwater image, the image will become darker. This is another problem that reduces the quality of underwater images because the light fades and the colors diminish [41]. To overcome these problems, pre-processing steps are needed to remove the noise from the underwater images.

Several researchers introduced different methods for image enhancement. Evolutionary algorithms such as genetic algorithm (GA) are used for improving images. The particle swarm optimization (PSO) algorithm is another evolutionary algorithm that is used to enhance gray-level images [18] and colored images [19]. Braik and Sheta [5] use the PSO algorithm for ordinary image enhancement. This approach was compared with the GA-based enhancement technique. The results showed that the PSO-based enhancement method is better in terms of time, number of pixels on edges, and the obtained objective scores. Because this enhancement model proved its superiority, we adopt the same approach here, namely the same enhancement model and objective function, but for underwater images. We perform pre-processing steps [3] to remove the noise from the underwater image, as this technique is designed to work for ordinary images.

Several methods have been introduced for the enhancement of underwater images, but to the best of our knowledge, evolutionary optimization algorithms are not used for underwater image enhancement. The quality of underwater images can be enhanced by improving the brightness, color correction, increasing the visibility, or enhancing scene contrast.

In the RGB model, the color components are not separated. This makes it unsuitable for enhancing images. Meanwhile, the hue, saturation, and value (HSV) model separates the image into three components,

***Corresponding author: Iyad Abu Doush,** Department of Computer Sciences, College of Information Technology and Computer Sciences, Yarmouk University, 21163 Irbid, Jordan, e-mail: eyad_win@yahoo.com
**Amal AbuNaser, Nahed Mansour and Sawsan Alshattnawi:** Computer Science Department, Yarmouk University, Irbid, Jordan

which are *H* (the color content), *S* (color saturation), and *V* (which is the luminance value of the color) [23]. HSV gives more flexibility in modifying the image. In our proposed technique, there is a conversion step for the image from the RGB model to the HSV model, and after applying the enhancement model, the image is converted back into RGB.

In this article, a framework based on the PSO algorithm is used to enhance the brightness, increase the visibility, and enhance the contrast of underwater images. The PSO algorithm is used in the literature to enhance gray-level images [18] and colored images [19]. We apply this algorithm to underwater images as follows: first, the PSO is used in RGB adjustment of the original underwater image. Then, the PSO is used to maximize a fitness function for calculating an intensity transformation function for the input image.

The proposed technique in this article is evaluated using *PSNR*, Kullback–Leibler divergence (*KLDIV*), number of edges, and a histogram. The results show that the proposed algorithm gives a good result, compared with that of Bakhtiari et al. [3], as it improves the illumination and the true colors of underwater images. In addition, the proposed technique is evaluated using mean value, standard deviation, and entropy.

The rest of the article is organized as follows: Section 2 presents the related work; the concept of PSO is presented in Section 3; the proposed framework is presented in Section 4; Sections 5 and Section 6 explain the results using the evaluation measurements; and finally, Section 7 concludes the work.

## 2 Related Work

There has been much research work done on the area of image enhancement. Iqbal et al. [21] use an unsupervised color correction method for improving low-quality images. Evolutionary algorithms for image enhancement has been used by Munteanu and Rosa [25], where they apply GA. Results are evaluated by comparing histogram equalization and linear contrast stretching. The results attained showed that this technique is better in both subjective and objective evaluations. The detailed variance and the background variance [33] were calculated. Bakhtiari et al. [3] present a color image enhancement method based on ensemble empirical mode decomposition (EEMD) and GA.

The PSO evolutionary algorithm has been used in the literature to enhance gray-level images [18] and colored images [19]. Gorai and Ghosh [18, 19] and Braik and Sheta [5] use the PSO algorithm for ordinary image enhancement; in the former, the enhanced images were better when compared with hue-preserving color image enhancement without the gamut problem and GA color image enhancement [18, 19]. The resulting images from the latter approach have been compared with GA-based enhancement technique [5]. The PSO-based enhancement method is better in terms of time, number of pixels on edges, and the objective scores obtained.

Many techniques are used for the enhancement of underwater images. Fairweather et al. [13] use techniques such as contrast stretching and Markov random field for image enhancement. Bimodal histogram model has been applied to enhance the underwater image.

In [20], an integrated color model has been used for underwater image enhancement. The quality of the images is statistically illustrated through the histograms. A histogram for the original image and the enhanced image is compared. The methods prove to enhance the underwater images.

Chambah et al. [7] presented a new algorithm for underwater image recognition in real time that is based on a combination of existing algorithms GW (Gray World), ACE (Automatic Color Equalization), and WP (Retinex White Patch). Cufi et al. [10] presented a vision-based system using motion detection. This approach is used to automatically maintain the position of the underwater motion vehicle when the reference of the corresponding image is lost. Gasparini and Schettini [16] developed a tunable cast remover depending on the modified version of the white balance algorithm. This approach uses a detector to specify the existence of a cast, then it works to remove it.

In another work [38], a physics-based model that concentrates on the recovery of the object is used. A polarizing filter to enhance the underwater images/scenes is applied. This approach concentrates on back scattering. Torres-Méndez and Dudek [41] analyze the color recovery, but from a different perspective. This approach uses Markov random field to model the image. The task of assigning suitable color values to the input image pixels is defined as the color correction process. This approach improves the underwater image color by building a probabilistic-based algorithm. It uses multiscale representations of the color-corrected and color-depleted images.

Bazeille et al. [4] proposed a pre-processing underwater image enhancement technique. This approach is used to enhance image quality by applying a group of independent steps. These processing steps make correction by non-uniform illumination (homorphic filtering), elimination of noise (wavelet denoising), enhancement of edges (anisotropic filtering), and adjustment of image colors (equalizing RGB channels to remove predominant color). The proposed algorithm is automatic and requires no parameter adjustment. The results are evaluated using the gradient magnitude histogram.

The slide stretching-based approach is proposed in reference [20]. This approach works two ways: the contrast stretching of the RGB algorithm is applied to equalize the color contrast in the images; then to solve the lighting problem, the saturation and intensity stretching of HSI is used. Prabhakar and Praveen Kumar [31] sequentially apply four filters (homomorphic filtering, wavelet denoising, bilateral filter, and contrast equalization) on the noisy image to produce an enhanced image.

Chiang et al. [8] presented a new algorithm that combines the dehazing algorithm and wavelength compensation to enhance underwater images. The dehazing algorithm is used to remove the haze effects from color scatter. After that, an estimation of the depth for the photography scene of each wavelength in the background light of the image is performed.

Anisotropic filtering and Kovesi's phase-preserving wavelet filtering are used for underwater images to improve edge detection [1]. A simple numerical value to assess the quality of the restoration procedure is presented. The results of this enhancement technique is evaluated qualitatively by finding the number of edges using the Canny–Deriche detector and quantitatively by calculating a numerical metric.

Ren et al. [35] presented a method to solve the illegibility problem of underwater digital images shot by underwater vision sensor by proposing the improved homomorphic filtering, which is based on mathematical morphology and uses differential evolution algorithm to optimize the parameters. Also, the proposed method was compared with the homomorphic filter image enhancement method based on Fourier transform and wavelet transform, using mean, standard deviation, and entropy as evaluation metrics to prove the superiority of proposed algorithm.

In [39], an approach using color correction based on histogram was used to improve visualization of underwater images. This approach was proposed to improve contrast by redistributing intensity distributions and computing a uniform histogram. The results are evaluated using mean, standard deviation, and median as evaluation metrics.

## 3 The PSO Algorithm

PSO is a population-based search algorithm that simulates the social behavior and the dynamic movement of birds within a flock [11]. Each bird in a search space adjusts its "flying" according to its own flying experience as well as the flying experience of other birds.

The population is initialized randomly with a group of particles [42], and each particle represents a solution. The algorithm searches for an optimum by a number of iterations. In each iteration, the particles are evaluated using a fitness function, and the value resulting from this function is called a particle fitness value. If the resulting particle fitness value is the best one, this particle stores the location of this value as the best value, personal best (*pbest*). At the end of each iteration, the particle with the best fitness value is selected as global best (*gbest*). Therefore, each particle keeps track of two values: its personal best (*pbest*) and the best value of any particle (*gbest*). These are responsible for guiding particles toward a better position [11]. Each particle adjusts its traveling speed by dynamically corresponding with the flying experiences of itself

```
P = Particle_Initialization();
For i=1 to it_max
  For each particle p in P do
    fp = f(p);
    If fp is better than f(pBest)
        pBest = p;
    end
  end
  gBest = best p in P;
  For each particle p in P do
      v = v + c1*rand*(pBest – p) + c2*rand*(gBest – p);
      p = p + v;
  end
end
```

**Figure 1.** The PSO Algorithm Pseudo-Code.

based on particle and its colleagues based on *gbest*. Therefore, the next position of any particle is modified according to:

1. its current position
2. its current velocity
3. the distance between its current position and *pbest*
4. the distance between its current position and *gbest*

The velocity ($V$) and position or location ($pL$) for each particle are updated using the following formulas [19]:

The velocity (V) equation:

$$V_i = wV_{i-1} + c_1 \times rand() \times (pbest_i - pL) + c_2 \times rand() \times (gbest - pL). \tag{1}$$

The position or location (pL) equation:

$$pL = pvL + V_i. \tag{2}$$

The inertia weight ($w$) can be calculated according to the following equation:

$$w = \frac{(T_{max} - t) - (w_{start} - w_{end})}{T_{max}} + w_{end}, \tag{3}$$

where $w$ is the inertia weight, $pvL$ is the location or position of the particle in the previous iteration, $pL$ is the current position for the particle, $V_i$ is the current velocity for particle $i$, $V_{i-1}$ is the previous velocity for particle $i$, $c_1$ and $c_2$ are the acceleration constants, $rand()$ is a random number, $pbest_i$ is the best value for particle $i$, and *gbest* is the best particle achieved over all iterations.

In the equation for inertia weight, $T_{max}$ is the maximum iteration, $w_{start}$ is the starting inertia weight, $w_{end}$ is the ending inertia weight, and $t$ is the current iteration.

The PSO algorithm steps could be summarized in Figure 1.

# 4 The Proposed Model

The PSO algorithm is applied to enhance the underwater images, and the algorithm steps are explained below and summarized in Figure 2.

**First Step:** RGB adjustment [3], which is the pre-process step. RGB represents the red, green, and blue channels. It is used in every computer system and television. Also, it is found in a system that displays images

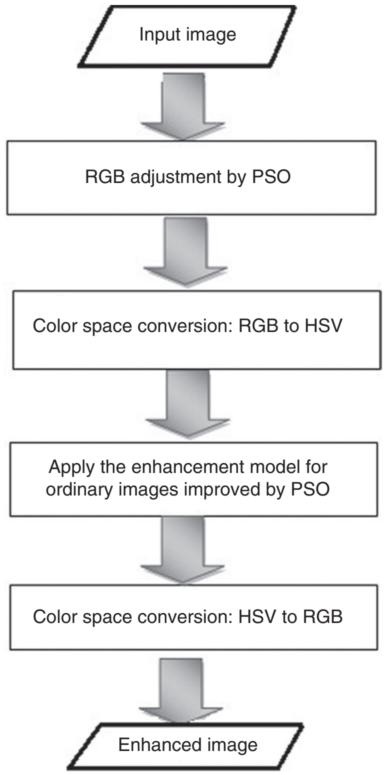


**Figure 2.** Block Diagram for the Underwater Image Enhancement Procedure.

using CRT. It is device dependent, and specification of colors is semi-intuitive [14]. In our proposed model, the RGB values of the input image should be modified in a way to make it more obvious by maximizing the information content in the input image. This step is carried out by PSO using a fitness function based on a target value. In the case of underwater images, the blue and green colors are dominant, so we need to adjust these channels to reduce the effect of water on the images before starting the enhancement process. Each channel of RGB should be adjusted to a desired mean value. To accomplish this, a constant value (positive or negative) is estimated by the PSO algorithm to adjust each channel. The RGB values are modified according to the following equation [3]:

$$R' = M1 + R, \tag{4}$$

$$G' = M3 + G, \tag{5}$$

$$B' = M2 + B, \tag{6}$$

where $R$, $G$, and $B$ are the original RGB values; $R'$, $G'$, and $B'$ are the corresponding values after modification; and $M1$, $M2$, and $M3$ are the constant values (positive or negative) that are used to modify the mean values of the red, green, and blue channels of the target values. These constant values are estimated by the PSO algorithm, and the goal is to find the best solution according to the fitness value.

The used fitness function depends on a target value, which is an array of three values: the first one for the red channel, the second one for the green channel, and the last one for the blue channel. The suitable value has been chosen after running the program several times on a group of images; in each trial, we change this target. The program continues running with different values for the target until the best enhancement is achieved.

In this program, a number of particles are initialized. Each particle has three components: $M1$, $M2$, and $M3$. The objective function (fitness value) is calculated for each particle in every iteration. When a termination criterion is reached, the best particle with the best fitness value is selected as the best solution for the RGB adjustment step.

**Second Step:** Color space conversion from RGB to HSV, which is carried out on the resulted image according to the following formulas [14]:

$$M = \max(R, G, B). \tag{7}$$

$$m = \min(R, G, B). \tag{8}$$

$$C = M - m. \tag{9}$$

$$H' = \begin{cases} \text{undefined} & \text{if } C = 0 \\ \frac{G-B}{C} mod6 & \text{if } M = R \\ \frac{B-R}{C} + 2 & \text{if } M = G \\ \frac{R-G}{C} + 4 & \text{if } M = B. \end{cases}$$
$$H = 60^\circ \times H'. \tag{10}$$

$$S_{HSV} = \begin{cases} 0 & \text{if } C = 0 \\ \frac{C}{V} & \text{otherwise}. \end{cases}$$
$$V = M. \tag{11}$$

The RGB model used is not efficient in the enhancement process because the color components are not separated. The HSV model separates the image into three components, which are $H$ (the color content), $S$ (color saturation), and $V$ (which is the luminance value of the color). The $H$ is kept intact and modifies either $S$ or $V$ or both [23]. HSV gives more flexibility in modifying the image.

Moreover, for human interpretation, the HSV model provides the exact description of the color than the RGB model. The HSV is user-oriented and has the best accuracy in color conversion [34].

The first two steps are necessary to deal with underwater image as an ordinary image.

**Third Step:** Applying the enhancement model for ordinary images (explained next) improved by PSO. The following enhancement transformation is applied to each pixel at location $(i, j)$ [25]:

$$g(i, j) = m(i, j)^a + [f(i, j) - c * m(i, j)]\left[k\frac{D}{\sigma(i, j) + b}\right], \tag{12}$$

where $f(i, j)$ and $g(i, j)$ are the gray-level intensity of pixels in the input and output images, respectively, centered at location $(i, j)$. The parameters $a$, $b$, $c$, and $k$ are to be taken as the same for the whole image and are defined over real positive numbers. $D$ is the global mean for the whole image and can be calculated using Equation (13) [18]; $m(i, j)$ is the gray-level mean computed for the neighborhood centered at $(i, j)$.

$$D = \left[\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} f(i, j)\right]\frac{1}{N \times M}. \tag{13}$$

$\sigma(i, j)$, the gray-level standard deviation computed at neighborhood centered at $(i, j)$, can be computed using Equation (14):

$$\sigma(i,j)=\sqrt{\frac{1}{n\times n}\Sigma_{x=0}^{n}\Sigma_{y=0}^{n}(f(x,y)-m(x,y))^{2}}. \tag{14}$$

PSO is used in this step with the goal of finding the best combination for the four parameters $a$, $b$, $c$, and $k$ according to an objective function shown in Equation (15) (i.e., $a$, $b$, $c$, and $k$ represent a particle where every particle represents a solution for the enhancement problem). In this proposed technique, the parameters are evaluated without human intervention using the objective function shown in Equation (15). This function concatenates multiple measures for performance: entropy (entropy value reveals the information content in the image), sum of edges intensities, and number of edges. It can give an overview about the quality of the enhanced image [18].

$$F(Z)=\log(\log(E(I(Z))))*\frac{n-edgels(I(Z))}{M*N}*H(I(Z)), \tag{15}$$

where $F(Z)$ represents the fitness function, $Z$ is the particle ($abck$), $I(Z)$ is the image after applying Equation (12), and the quantity ($E(I(Z))$) represents the intensity of the edges detected with any edge detector algorithm applied to the transformed image $I(Z)$ using Equation (12). Edge detection can be done using Sobel [17], Laplacian [17], and Canny [6] edge detectors. Here, Sobel [17] is used as an automatic threshold detector [37]. $n–edgels(I(Z)$ is the number of pixels whose intensity value is above a threshold in the Sobel edge image. $H$ is the entropy value. $M$ and $N$ represent the size of the image given by the number of pixels in the horizontal and vertical directions of the image, respectively. $E(I)$ represents the sum of intensities of the edges included in the image $I(Z)$ after applying Equation (12) [9].

**Fourth Step:** In the HSV model, $H$ denotes the hue, which represents the color content, $S$ represents the color saturation, and $V$ is the luminance value (brightness) of the color. The proposed algorithm enhances the true colors and increase the luminance of the image without modifying the original color contents. After getting the enhanced image, the $S$ and $V$ components, which are enhanced, together with the unaltered $H$ component, are converted back to the RGB color image. Then, a color space conversion from HSV to RGB is carried out, which is an opposite conversion for the third step. This conversion is calculated according to the following formulas [14]:

$$C=V\times S_{HSV}. \tag{16}$$

$$H'=\frac{H}{60^{\circ}}. \tag{17}$$

$$X=C(1-|H'mod2-1|). \tag{18}$$

$$(R_1,G_1,B_1)=\begin{cases}(0,0,0) & \text{if } H \text{ is undefined}\\ (C,X,0) & \text{if } (0\le H'<1)\\ (X,C,0) & \text{if } (1\le H'<2)\\ (0,C,X) & \text{if } (2\le H'<3)\\ (0,X,C) & \text{if } (3\le H'<4)\\ (X,0,C) & \text{if } (4\le H'<5)\\ (C,0,X) & \text{if } (5\le H'<6).\end{cases}$$

$$m=V-C. \tag{19}$$

$$(R,G,B)=(R_1+m,B_1+m). \tag{20}$$

# 5 Experiments

The proposed algorithm has been implemented using MATLAB and tested on a collection of underwater images. First, the program has been run for 100 iterations and 50 particles. Then, the evaluation measures are used. After that, we rerun the program after increasing the number of iterations to 150 and use the evaluation measures. We continue to increase the number of iterations and the number of particles until we get the best results by setting the number of particles to 200 and the number of iterations to 200.

In the RGB adjustment step, we use a fitness function based on a target value. This target value has been set to the proper value after running the program several times on a group of images. Then we choose the target value that gives the best enhancement.

We have tested our proposed technique on four color images taken from [3]. All these images were taken at marine habitats. The camera was too far away from the scene. Images has too little contrast and are dull, with a heavy cyan cast. Image 1 was taken at a marine habitat. Image 2 was taken at the Alam Anda house reef, Sambirentreng, Bali, marine habitats (setting: ISO 100, F2, 8, 1/320s, focal length 8.3 mm, flash to half power). Image 3 was taken at the wreck of the USAT *Liberty* in Tulamben, Bali (setting: ISO 100, F5, 1/125s, focal length 6.6 mm, flash to half power). Image 4 was taken at a marine habitat [43].

The program has been run using Microsoft Windows XP professional with Intel Core 2 Duo, CPU 2.00 GHz, and with 2 GB of RAM.

# 6 Evaluation

The evaluation is done using seven different metrics, but first we discuss the initialized values for the algorithm parameters.

## 6.1 Parameter Setting

The results for the PSO algorithm depends on choosing suitable values for the parameters. In this method, parameters $c_1$ and $c_2$, the acceleration values, are set to 1.4, *rand*() is a random number between [0,1] and it is different in every generation for each particle components.

The range for $a$, $c$, and $k$ parameters are the same as [26]. $a \in [0, 1.5]$, $b \in [0, 0.5]$, $c \in [0, 1]$, and $k \in [0.5, 1.5]$. The proposed range for parameter $b$ in [26] did not produce good results. To obtain better results, we used the value $b \in [0, GlobalMean/2]$, which is the same as in [19].

The inertia weight $w$ is reduced linearly, from *wstart* to *wend*, for each iteration. In this method, the values for *wstart* is set to 0.9 and *wend* is set to 0.4.

## 6.2 Evaluation Criteria

Many evaluation methods are used to evaluate underwater image enhancement techniques. Computable distortion measures such as mean squared error, signal-to-noise ratio, and *PSNR* have been widely calculated and used for the evaluation. In [28], a comparison of filters used for underwater image pre-processing in terms of mean square error (*MSE*) and *PSNR* is used. The *PSNR* value is calculated as follows:

$$PSNR = 20 \log 10(256 / \sqrt{MSE}), \tag{21}$$

where *MSE* is the mean square error of the estimation.

The *PSNR* can be used [31] for evaluating underwater image enhancement technique. This measure is used along with other quantitative-based criteria such as gradient magnitude histogram. The proposed technique achieved the highest *PSNR* compared with other techniques. Higher values of *PSNR* are better because it means that the ratio of the signal (the image after enhancement) to the noise (which is the original image) is higher [32].

A comparison between image-filtering algorithms can be performed using several measures [27]. Two of the widely used measures are the *PSNR* and *MSE*. We used these two metrics to evaluate the obtained results.

Another metric used for image enhancement evaluation is the number of edges detected in the enhanced image compared with the original image. An edge in an image is the border between two adjacent areas where a set of connected pixels are found [17]. Edge detection is an important technique in image processing and feature extraction [15]. This metric is used by other researchers (e.g., Iqbal et al. [21]) to compare the results of the proposed technique with other existing methods.

There are several edge detection algorithms based on estimating the changing of transitions in gray levels in an image. One example of an edge detection algorithm is the Sobel edge detector. It is applied on both original and enhanced images [31].

The *KLDIV* can be defined as a non-symmetric measure of the difference between two probability density functions [12, 40]. The *KLDIV* is used in our work to evaluate the performance of the proposed method compared with the other method. The separation of two distributions can be quantified using the *KLDIV* [12]. This measure is non-negative, and when its value is close to zero, the comparable methods are similar (i.e., it is hard to distinguish between them) [12].

This metric is calculated according to the following formula [30]:

$$KLS(P||Q) = KL(P||Q) + KL(Q||P), \tag{22}$$

where

$$KL(P\,||\,Q) = sumP(x)\log\frac{p(x)}{Q(x)}. \tag{23}$$

Using the *KLDIV*, the separation between two distributions can be simply seen as a scalar value that gives performance overview for multiple operating points at the same time [12, 22].

Histogram equalization is carried out by comparing two histograms: the input image and the enhanced image. According to Iqbal et al. [21], "[t]he wider histogram represents a more visually appealing image." Many researchers use this measure to evaluate the performance of their methods (e.g., [36]).

The mean value reflects the mean color of image. The mean value of the enhancement image should differ from the mean value of the input image [35, 39].

The standard deviation refers to the image details. As the standard deviation becomes larger, rich details appears on the image [39]. The mean value and standard deviation metrics are used by Ren et al. [35] and by Shamsuddin et al. [39].

Entropy reflects the information of the image. As the entropy value increases, the quality of the image gets better [35]. A higher entropy indicates that the image contains bountiful information. The entropy metric is used by Ren et al. [35].

Other methods have been used for evaluating enhancement techniques, as can be seen in [3]. The input image and enhanced image can be compared by a human viewer.

The proposed technique in this article has been evaluated using *PSNR*, *KLDIV*, number of edges, histograms, mean, standard deviation, and entropy. The results show that the proposed algorithm gives good results because it improves the illumination and true colors of underwater images. The original and enhanced underwater images after applying the PSO algorithm are shown in Figure 3.

## 6.3 Result Analysis

This proposed method has been tested on four color images taken from [3]. The images are enhanced by EEMD and GA, but the resulting images were evaluated subjectively by a human viewer. They did not use any known evaluation method. We obtained from Bakhtiari et al. [3] the input images and the output images

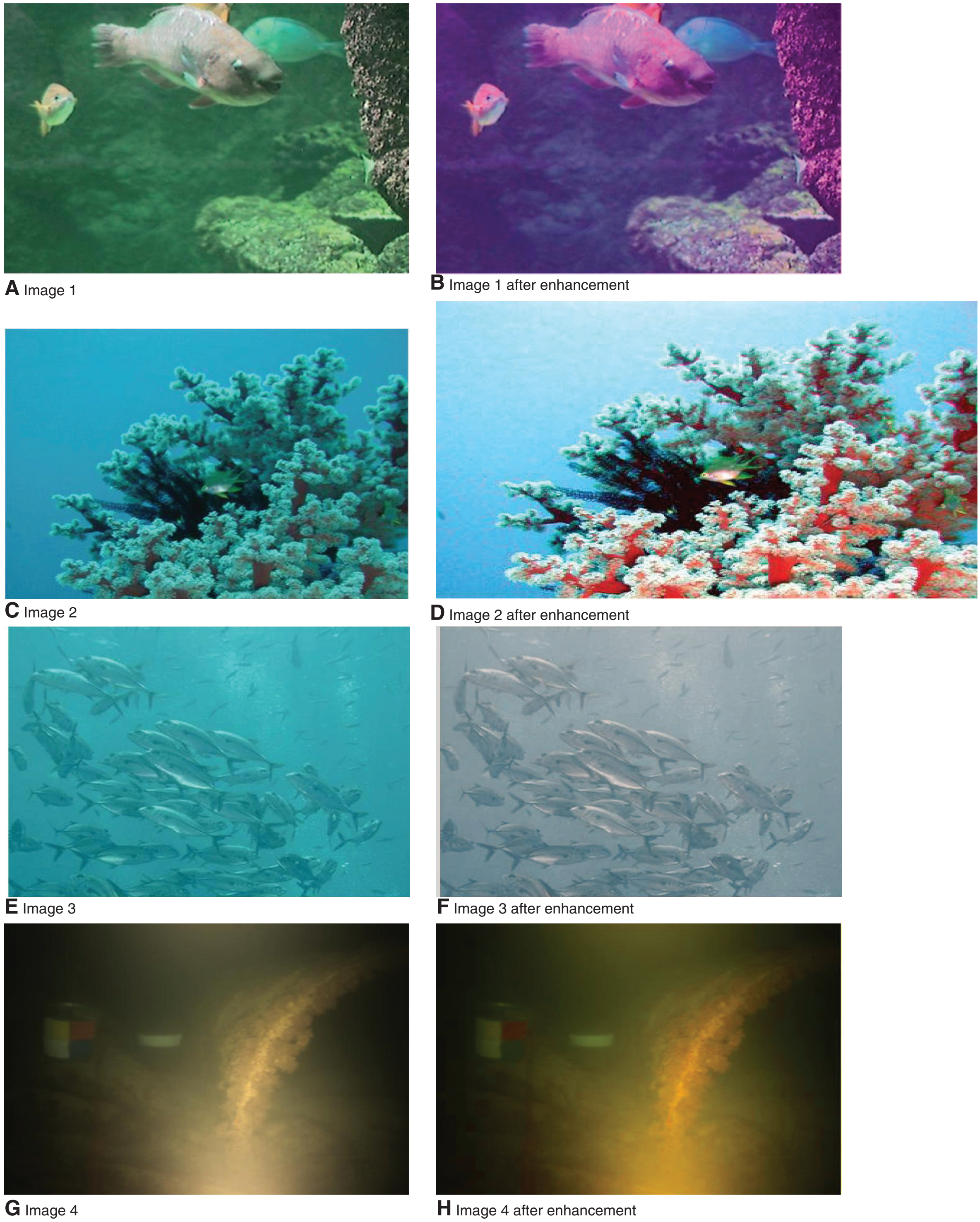


**Figure 3.** The Original and Enhanced Underwater Image after Applying the PSO Algorithm.

after applying their algorithm. We applied the same evaluation metrics that have been used for our proposed technique on the output images for the proposed method to compare the results.

The results are evaluated using the following:

1. *PSNR*: a computable distortion measure. The results for the *PSNR* value are shown in Table 1, which are from a code given in [29]. A higher *PSNR* value indicated a better enhancement result because it means

**Table 1.** *PSNR* for the Four Test Images.

| Image | *PSNR* for PSO | *PSNR* for EEMD+GA |
|---|---|---|
| Image 1 | 59.0615 | 43.045 |
| Image 2 | 53.0198 | 46.6815 |
| Image 3 | 48.1100 | 45.8139 |
| Image 4 | 42.0427 | 41.6052 |

that the ratio of the enhanced image (signal) to the original image (noise) is higher. Therefore, the results in the table indicate that the images were better enhanced using PSO than EEMD and GA. From Table 1, the noise ratio of image 1 is 59.0615, which is greater than the noise ratio for the EEMD and GA; also, the noise ratio for image 2, image 3, and image 4 gives greater values using PSO than noise ratio values using EEMD and GA.

2. *KLDIV*: a non-symmetric measure of the difference between two probability density functions [12]. The results for this measure are shown in Table 2, which are from a code given in [24]. The value of this measure must be >0 and <1. Table 2 shows that the results are all between 0 and 1, and the *KLDIV* value for image 1 is 0.743, which means that the difference between image 1 after enhancement and the original image is 0.743; for image 2, the difference is 0.631; and the *KLDIV* value for image 3 is 0.576, which shows that the image is enhanced; also, image 4 satisfies a good enhancement in which the difference between the enhanced image and the input one is 0.595. This metric indicates that using PSO enhances the images and reduces the similarity between the input image and the enhanced one. These values are higher than the values obtained by the EEMD and GA, which means that PSO is more powerful.
3. Number of detected edges: an edge in an image can be defined as a boundary between two adjacent regions where a set of connected pixels are found. If the number of edges for the enhanced image is greater than the number of edges for the original image, then the image is enhanced. The greater the number of edges, the more enhanced image is produced. The edge detection algorithm that is used here is the Sobel edge detector. As shown in Table 3, the number of edges for enhanced image using PSO is greater than the number of edges using EEMD and GE.
   When comparing our results using PSO with the results using EEMD and GA, we can infer the following: from Table 3, the number of edges for original image 1 is 6246 and for the enhanced image 1 using PSO, 6996, and the difference is 750, which shows a better image with a larger number of edges, whereas the

**Table 2.** Results for *KLDIV*.

| Image | *KLDIV* for PSO | *KLDIV* for EEMD+GA |
|---|---|---|
| Image 1 | 0.7430 | 0.0570 |
| Image 2 | 0.6310 | 0.4827 |
| Image 3 | 0.5768 | 0.4019 |
| Image 4 | 0.5254 | 0.0983 |

**Table 3.** Number of Edges Detected Using Sobel Edge Detector for the Four Test Images.

| Image | Original image | Enhanced images by PSO | Enhanced images by EEMD+GA |
|---|---|---|---|
| Image 1 | 6246 | 6996 | 6246 |
| Image 2 | 10,182 | 12,383 | 10,182 |
| Image 3 | 6193 | 8540 | 6193 |
| Image 4 | 543 | 680 | 543 |

number of edges for the enhanced image 1 using EEMD and GA is 6246, which is the same as the number of edges in the original image 1; this is because EEMD and GA did not enhance the number of edges. The same applies for the other images. We apply this test for the images that we obtained from the authors of the EEMD and GA article; the authors did not apply this test, and this means that the used fitness function does not handle the number of edges.

4. Histogram figures: used to evaluate the quality of an image. The wider histogram represents better results. The figures below indicate the quality of the images by comparing the input image histogram with the enhanced image histogram. The enhanced histograms are stretched on a wider range than the histograms of the original images. This is shown in Figure 4.
   Also, when comparing the histogram figures after using PSO, as shown in Figure 4, with the histogram figures after using EEMD and GA, as shown in Figure 5, we can infer that the histogram figures for the enhanced images using PSO are stretched on a wider range than for the histograms for the enhanced images using EEMD and GA.

Using these four evaluation methods and by comparing the image after enhancement using PSO with the image after enhancement using EEMD and GA, we conclude that applying the PSO algorithm for underwater images is better than the recently used algorithms.

Also, the results are evaluated using the following:

1. Mean value: reflects the mean color of image and is used to measure the amount of improvement in the image; the image is enhanced if there is a difference in terms of color between the original image and the enhanced one. The results for this measure are shown in Table 4. The mean value for original image 1 is 94.1996, and the mean value for the enhanced image 1 is 101.9123, which means that the difference between image 1 after enhancement and the original image is 7.7127, the difference for image 2 is 12.8471, and the difference for image 3 is 21.3087, which shows that the images are enhanced; also, image 4 satisfies a good enhancement in which the difference between the enhanced image and the input one is 14.8338. This metric indicates that using PSO enhances the images.
2. Standard deviation: refers to the image details. As the standard deviation becomes larger, the image appears with rich details [35]. The results for this measure are shown in Table 5. Table 5 shows that the standard deviation increased for the enhanced images after applying our proposed method compared with the values for the original ones, which means that the enhanced images contain more details.
   Ren et al. [35] propose improved homomorphic filtering based on mathematical morphology and use differential evolution algorithm to optimize the parameters to solve the illegibility problem of underwater digital images. The evaluation methods that have been used are mean, standard deviation, and entropy.
   Shamsuddin et al. [39] propose an approach to improve the visualization of underwater images using color correction based on histogram. They evaluate this approach using mean, standard deviation, and median.
   In [35, 39], the evaluation metrics were assessed by calculating these values, and if there was a difference between the values of the original images with the corresponding values for the enhanced images according to the definition of each measure, this would mean that the proposed method achieve good enhancement.
   If we compare the differences in mean value and standard deviation for our proposed method with the corresponding differences for other approaches, we can infer that the proposed method satisfies good enhancement. In [35], the differences in mean value for their own image was 17.682, and in [39], the differences in mean value for their own images vary from 4.79 to 22.9, whereas in our proposed method, the range for differences in mean values was from 7.7127 to 21.3087.
   Regarding the comparison of the standard deviations, in [35], the increment in the standard deviation for their own image was 5.494, and in [39], the increment in the standard deviation ranged from 0.07 to 6.43. In our proposed approach, the increment in standard deviation was between 3.77 and 9.721.
3. Entropy: reflects the information of the image. As the entropy value increases, the quality of the image increases [35]. This measure is similar to the standard deviation in that a higher entropy indicates that

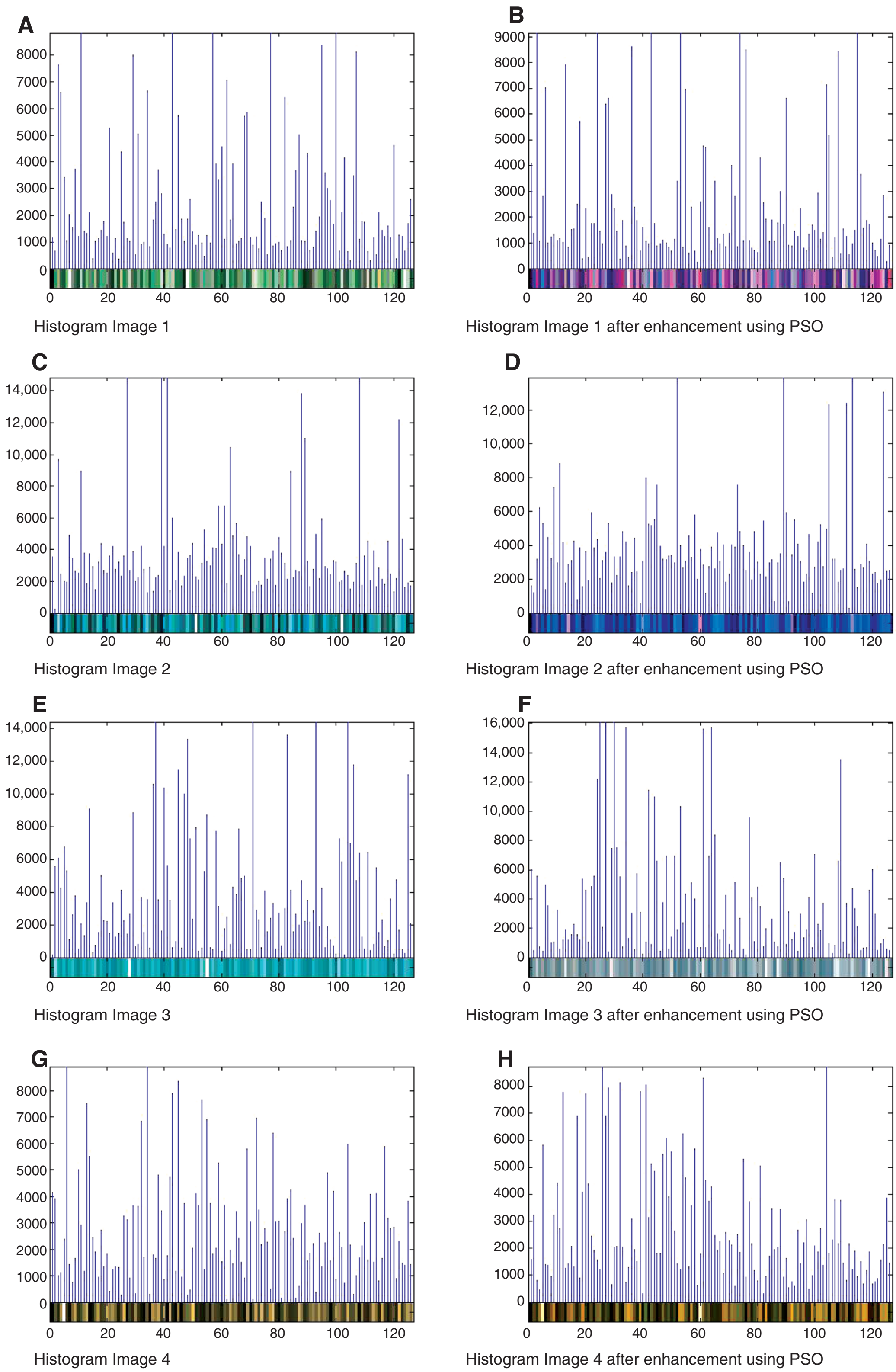


**Figure 4.** The Original and Enhanced Underwater Image Histograms Using PSO.

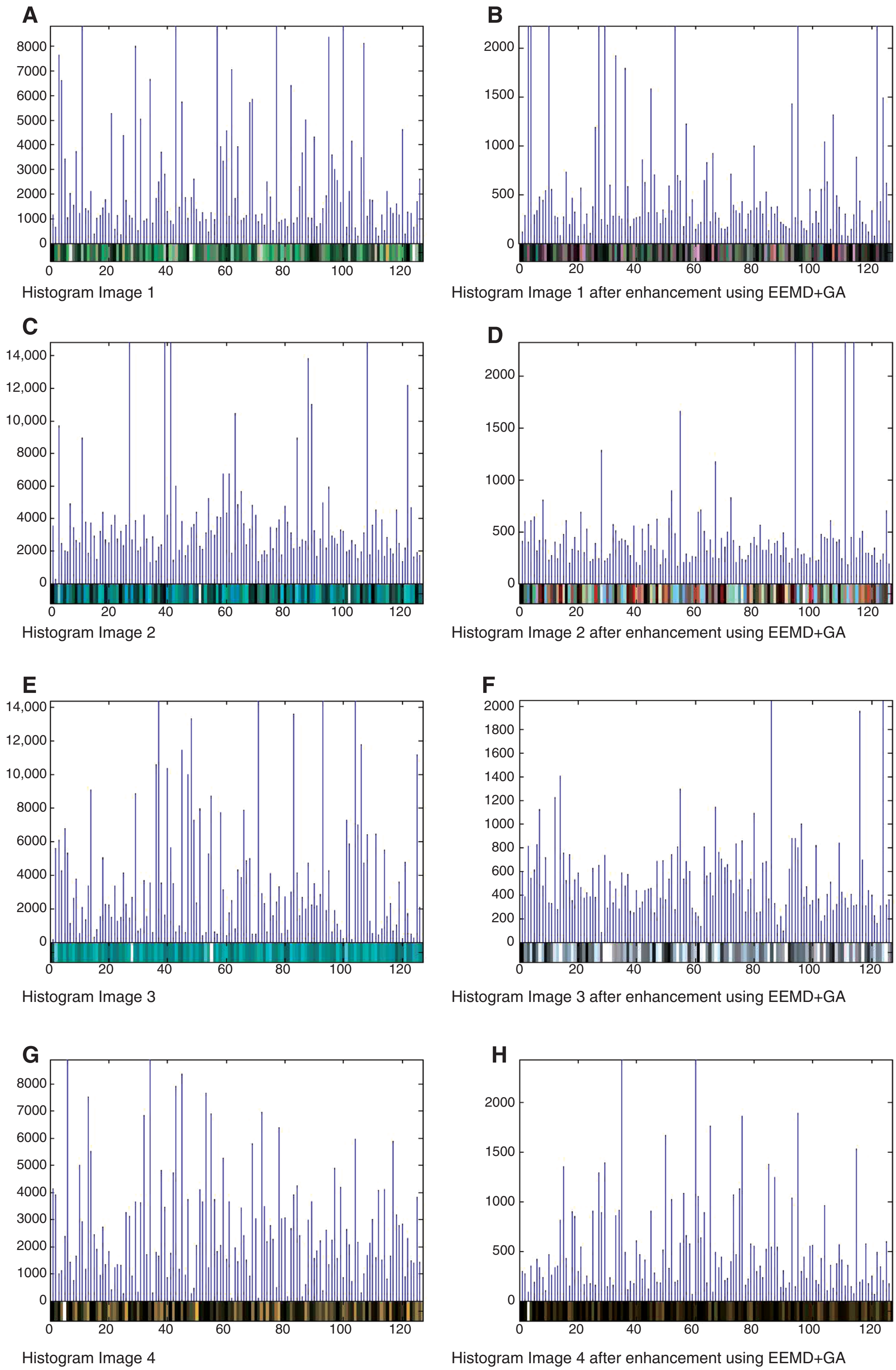


**Figure 5.** The Original and Enhanced Underwater Image Histograms using EEMD+GA.

**Table 4.** Results for Mean.

| Image | *Mean* for original image | *Mean* for enhanced image |
|---|---|---|
| Image 1 | 94.1996 | 101.9123 |
| Image 2 | 97.3591 | 110.2462 |
| Image 3 | 125.3125 | 146.6212 |
| Image 4 | 46.9295 | 61.7633 |

**Table 5.** Results for Standard Deviation (*Std*).

| Image | *Std* for original image | *Std* for enhanced image |
|---|---|---|
| Image 1 | 43.9811 | 48.0335 |
| Image 2 | 62.3166 | 72.0370 |
| Image 3 | 55.4620 | 59.2378 |
| Image 4 | 43.4411 | 47.7058 |

**Table 6.** Results for Entropy.

| Image | *Entropy* for original image | *Entropy* for enhanced image |
|---|---|---|
| Image 1 | 7.2314 | 7.8444 |
| Image 2 | 7.5833 | 7.9622 |
| Image 3 | 7.0437 | 7.7703 |
| Image 4 | 7.1105 | 7.5010 |

the image contains bountiful information. Table 6 shows the results of this measure. Table 6 illustrates that the enhanced images contain abundant information in comparison with the corresponding original ones. Also, this metric shows an acceptable enhancement compared with another approach (e.g., Ren et al. in [35]).

In [35], the increment in the entropy was 0.6; in our approach, the amount of increment ranged from 0.3789 to 0.7266, which indicates that our proposed approach satisfies good enhancement.

## 7 Conclusion

In this article, a PSO algorithm-based underwater image enhancement framework has been proposed. The implementation and testing are realized to prove the efficiency of this algorithm. The main goal of this technique is to enhance the image as much as possible by testing some parameters and comparing the results with another technique used.

One of the most important parameters that must be checked is the number of edges, which helps in the observation of more details in the image, and the goal is always to maximize it without human intervention. To achieve the best results, an appropriate fitness function that combines the parameters intensity, number of edges, entropy measure, standard deviation, and mean value of pixels has been chosen, implemented, and tested. In the state of the art, each fitness function concentrates on some parameters and ignores others. For example, the function chosen by the EEMD and GA does not take into consideration the number of edges.

The proposed algorithm, along with the fitness function, efficiently improves the underwater images, i.e., the illumination and true colors. The technique is tested on four selected underwater images using seven evaluation methods. The results obtained are tabulated and histograms are plotted, indicating that the proposed technique enhances underwater images based on the obtained results themselves and by comparing the results with the most recently proposed algorithm (EEMD and GA).

## Bibliography


[1] A. Arnold-Bos, J. P. Malkasse and G. Kervern, A preprocessing framework for automatic underwater images denoising, in: *Proceedings of the European Conference on Propagation and Systems* 2005, Brest, France, 2005.

[2] A. Arnold-Bos, J. P. Malkasse and G. Kervern, Towards a model-free denoising of underwater optical images, in: *Oceans 2005* – Europe, vol. 1, pp. 527–532, 2005.

[3] S. Bakhtiari, S. Agaian and M. Jamshidi, A color image enhancement method based on ensemble empirical mode decomposition and genetic algorithm, in: *World Automation Congress (WAC)*, 2012, pp. 1–6, 2012.

[4] S. Bazeille, I. Quidu, L. Jaulin and J. Malkasse, Automatic underwater image pre-processing, in: *Proceedings of the Caracterisation du Milieu Marin* (CMM '06), 2006, 1–8.

[5] M. Braik and A. Sheta, Image enhancement using particle swarm optimization, in: *Lecture Notes in Engineering and Computer Science*, 2007, 696–701.

[6] F. Canny, A computational approach to edge detection, *J. IEEE Pattern Anal. Machine Intell.* **8** (1986), 679–698.

[7] M. Chambah, D. Semani, A. Renouf, P. Courtellemont, A. Rizzi and L. Rochelle, Underwater color constancy: enhancement of automatic live fish recognition, in: *SPIE/IS&T Electronic Imaging* 2004, 2004.

[8] J. Chiang, Y. C. Chen and Y. F. Chen, Underwater image enhancement: using wavelength compensation and image dehazing (WCID), in: *Advanced Concepts for Intelligent Vision System*, vol. 6915, pp. 372–383, Springer-Verlag, Berlin, Heidelberg, 2011.

[9] L. Coelho, J. Sauer and M. Rudek, Differential evolution optimization combined with chaotic sequences for image contrast enhancement, *Chaos Solitons Fractals* **42** (2009), 522–529.

[10] X. Cufi, R. Garcia and P. Ridao, An approach to vision-based station keeping for an unmanned underwater vehicle, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, vol. 1., pp. 799–804, 2002.

[11] Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez and R. G. Harley, Particle swarm optimization: basic concepts, variants and applications in power systems, *IEEE Trans. Evol. Comput.* **12** (2008), 171–195.

[12] D. Eriksson, L. Eriksson, E. Frisk and M. Krysander, Analysis and optimization with the Kullback–Leibler divergence for misfire detection using estimated torque, Technical Report LiTH-R-3057, Department of Electrical Engineering, Linköpings Universitet, Linköping, Sweden, 2013.

[13] A. Fairweather, M. Hodgetts and A. Greig, Robust scene interpretation of underwater image sequences, in: *6th International Conference on Image Processing and Its Applications*, pp. 660–664, 1997.

[14] A. Ford and A. Roberts, *Colour space conversions*, 1998 (technical report ).

[15] W. Gao, X. Zhang, L. Yang and H. Liu, An improved Sobel edge detection, in: *3rd IEEE International Conference on Computer Science and Information Technology* (ICCSIT 2010), vol. 5, pp. 67–71, 2010.

[16] F. Gasparini and R. Schettini, Color correction for digital photographs, in: *Proceedings of the 12th International Conference on Image Analysis and Processing*, ICIAP '03, IEEE Computer Society, pp. 646–651, Washington, DC, 2003.

[17] R. Gonzales and R. E. Woods, *Digital Image Processing*, Upper Saddle River, NJ: Pearson Prentice Hall, 2001.

[18] A. Gorai and A. Ghosh, Gray-level image enhancement by particle swarm optimization, in: *World Congress on Nature Biologically Inspired Computing, 2009*, NaBIC 2009, pp. 72–77, 2009.

[19] A. Gorai and A. Ghosh, Hue-preserving color image enhancement using particle swarm optimization, in: *2011 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, pp. 563–568, 2011.

[20] K. Iqbal, A. Rosalina, A. Osman and A. Talib, Underwater image enhancement using an integrated colour model, *IAENG Int. J. Comput. Sci.* **34** (2007), 239–244.

[21] K. Iqbal, M. Odetayo, A. James, R. Salam and A. Talib, Enhancing the low quality images using unsupervised colour correction method, in: *Systems 2010 IEEE International Conference on Man and Cybernetics (SMC)*, pp. 1703–1709, 2010.

[22] D. Johnson and S. Sinanovic, Symmetrizing the Kullback–Leibler distance, Technical Report, *IEEE Trans. Inform. Theory* (2000), 1–10.

[23] N. Kansal, *Fuzzy techniques for image enhancement*, Ph.D. thesis, Thepar University, Patiala, Punjab, India, 2010.

[24] Kullback–Leibler divergence, World Wide Web electronic publication (2007). http://www.mathworks.com/matlabcentral/fileexchange/13089-kldiv/content/kldiv.m.

[25] C. Munteanu and A. Rosa, Towards automatic image enhancement using genetic algorithms, in: *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 2., pp. 1535–1542, 2000.

[26] C. Munteanu and A. Rosa, Gray-scale image enhancement as an automatic process driven by evolution, *IEEE Trans. Syst. Man Cybern. B* **34** (2004), 1292–1298.
[27] G. Padmavathi, P. Subashini, M. Kumar and S. Thakur, Performance analysis of non linear filtering algorithms for underwater images, *IJCSI* **6** (2009), 232–238.
[28] G. Padmavathi, P. Subashini, M. Kumar and S. Thakur, Comparison of filters used for underwater image pre-processing, *Int. J. Comput. Sci. Network Secur.* **10** (2010), 58–64.
[29] Peak signal to noise ratio. World Wide Web electronic publication (2012). http://www.mathworks.com/matlabcentral/fileexchange/37691-psnr-for-rgb-images/content/PSNR_RGB.m.
[30] J. Perkio and A. Hyvarinen, Modelling image complexity by independent component analysis, with application to content-based image retrieval, in: C. Alippi, M. M. Polycarpou, C. G. Panayiotou and G. Ellinas, eds., *ICANN (2), Lecture Notes in Computer Science*, pp. 704–714, Springer-Verlag, Berlin, Heidelberg, vol. 5769, 2009.
[31] C. J. Prabhakar and P. U. Praveen Kumar, An image based technique for enhancement of underwater images, *Int. J. Mach. Inte.* **3** (2011), 217–224.
[32] T. Prabhakar, V. Naveen, A. Prasanthi and G. Santhi, Image compression using dct and wavelet transformations, *Int. J. Signal Process. Image Process. Pattern Recognit.* **4** (2011), 61–74.
[33] G. Ramponi, N. Strobel, S. Mitra and T. H. Yu, Nonlinear unsharp masking methods for image contrast enhancement, *J. Electron. Imaging* **5** (1996), 353–366.
[34] C. Ren and J. Yang, A novel color microscope image enhancement method based on HSV color space and curvelet transform, *IJCSI* **9** (2012), 272–277.
[35] Y. Ren, J. Li and W. Bi, A study on underwater sensing image enhancement based on differential evolution, *Sensor Lett.* **10** (2012), 1552–1556.
[36] A. Rizzi, C. Gatta and D. Marini, A new algorithm for unsupervised global and local color correction, in: *Colour Image Processing and Analysis. First European Conference on Colour in Graphics, Imaging, and Vision* (CGIV 2002), *Pattern Recognit. Lett.* **24** (2003), 1663–1677.
[37] P. Rosin, Edges: saliency measures and automatic thresholding, *Mach. Vision Appl.* **9** (1997), 139–159.
[38] Y. Y. Schechner and N. Karpel, Clear underwater vision, in: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR 2004, vol. 1, pp. 536–543, 2004.
[39] N. B. Shamsuddin, W. F. B. Wan Ahmad, B. Baharudin, M. K. B. Mohd Rajuddin and F. B. Mohd, Image enhancement of underwater habitat using color correction based on histogram, in: *IVIC (1), Lecture Notes in Computer Science*, pp. 289–299, Springer-Verlag, Berlin, Heidelberg, vol. 7066, 2011.
[40] H. Talebi and N. Esmailzadeh, Using Kullback–Leibler distance for performance evaluation of search designs, *Bull. Iran. Math. Soc.* **37** (2011), 269–279.
[41] L. Torres-Méndez and G. Dudek, Color correction of underwater images for aquatic robot inspection, in: A. Rangarajan, B. Vemuri and A. Yuille, eds., *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, vol. 3757, pp. 60–73, 2005.
[42] I. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, *Inform. Process. Lett.* **85** (2003), 317–325.
[43] Underwater images, World Wide Web electronic publication (2002). http://www.dive.snoack.de/tutorials/g_Tutorial_01.html#Photo1a.