Abstract

Background/Objectives: The major aim of the work is to propose an efficient multi-level thresholding for gray scale image using Firefly Algorithm (FA). Methods/Statistical Analysis: The multi-level image thresholding is attempted using Otsu’s function and Firefly Algorithm (FA) using standard 512 x 512 sized gray scale image dataset. The robustness of the attempted segmentation process is tested by staining the test images with universal noises. The superiority of the FA based segmentation is validated with the heuristic algorithms, such as Bat Algorithm, Bacterial Foraging Optimization and Particle Swarm Optimization existing in the literature. Findings: The simulation result in this work conforms that, FA assisted segmentation offers better result compared to the alternatives. The robustness of the FA and Otsu based segmentation is also superior and offered improved cost function, SSIM, PSNR value and reduced CPU time compared with the alternatives. Application/Improvements: In future, the proposed technique can be experienced using standard RGB images available in the literature.

Keywords: Firefly Algorithm, Multithresholding, Noise, Otsu, Performance Measure, Test Images

1. Introduction

In imaging science, image processing plays an essential role in the analysis and interpretation of images in various fields, such as medical discipline, damage detection, texture and pattern recognition, and. The advancement of digital imaging procedures and computing technology improved the budding of imaging science. Segmentation process is one of pre-processing procedure extensively employed in imaging field to extract key features from test data. Numerous methods for image processing have been discussed and executed in literature. In segmentation, the input image is separated into non-overlapping and homogenous groups having similar objects. The methods such as Kapur, Tsallis, and Otsu are widely used by great number of researchers to find solution for multilevel image segmentation problems.

In recent years, significant number of heuristic algorithm based segmentation procedures are developed and implemented by most of the researchers. Sathya and Kayalvizhi proposed a BFO based approach to segment the gray level images. This approach discovers the most favourable thresholds by the entropy and between-class variance based cost function. Sathya and Kayalvizhi also proposed similar thresholding procedures for various gray scaled images with traditional and modified forms of heuristic algorithms.

Manikantan et al. discussed Golden Ration PSO based approach to determine optimal thresholds for gray scale images based on Tsallis entropy method. They applied this segmentation approach on Lena, pepper, baboon, cameraman, and aeroplane images and the proposed method is validated with GA, BFO and PSO.

Akay presented a detailed multi-level segmentation procedure with Otsu’s and Kapur’s functions. In this study, heuristic search is considered to find the best threshold value. The well known image parameters are considered to judge the image eminence.

Oliva et al. discussed an image thresholding
methodology using, a multi-level thresholding method using the Harmony Search Algorithm\textsuperscript{14}. Quality of image is assessed by considering the cost functions, such as Otsu’s or Kapur’s methods and concluded that, HSA based method demonstrate the high performance for the segmentation of digital images. Raja et al. proposed the histogram based multi-level thresholding approach using Brownian Distribution (BD) guided FA\textsuperscript{15}. They also suggested an enclosed search method to maximize the optimization precision with lesser search iterations. Otsu’s function is maximized to attain optimal threshold level for gray scale images.

From the literature, it can be observed that, Otsu based thresholding procedure established for their better silhouette and consistency measures for multi-level thresholding works. Due to the increased complexity in the multi-level thresholding work, softcomputing algorithms are extensively considered by the researchers\textsuperscript{15–18}.

In this work, gray scale image multi-thresholding is carried out with the Otsu and FA. To analyse the robustness in the anticipated method, the test images are stained with the noise values. The segmentation process is then tested on the noise stained test images and the result shows that, FA based segmentation offers enhanced outcome compared with BA, PSO and BFO existing in the literature. The experimental job is achieved by the MATLAB 7.0 and the result shows that, FA based segmentation offers robustness in the anticipated method, the test images are carried out with the Otsu and FA.

In this work, most common image performance measure values are considered as given below\textsuperscript{15,16}.

The PSNR is defined as:

\[
PSNR(a,b) = 20 \log_{10} \left( \frac{255}{\sqrt{MSE(a,b)}} \right)
\]  

where, \(a=\)original image and \(b=\)segmented image.

The SSIM is generally considered to guess the picture quality and inter-dependency among images\textsuperscript{10,19}.

\[
SSIM(a,b) = \frac{(2\mu_a\mu_b + C_1)(2\sigma_{ab} + C_2)}{\mu_a^2 + \mu_b^2 + C_1(\sigma_a^2 + \sigma_b^2 + C_2)}
\]

where \(\mu_a\) and \(\mu_b\) are the means of the images \(a\) and \(b\), \(\sigma_a^2\) and \(\sigma_b^2\) are their variances, \(\sigma_{ab}\) is the covariance of \(a\) and \(b\), and \(C_1\) and \(C_2\) are stabilizing constants.

\section{3. Heuristic Algorithms Considered in this Work}

In this manuscript, Otsu’s function based segmentation procedure is initially attempted using the Firefly Algorithm based approach recently discussed in\textsuperscript{15}. The outcome of this method is then validated using the most successful heuristic methods, like BA, BFO and PSO.

\subsection{3.1 Firefly Algorithm}

Firefly algorithm is also proposed by Yang\textsuperscript{20} in 2009. It is created by reproducing the irregular lighting guidance formed by firefly. Detailed description and working principle of the firefly algorithm can be found in\textsuperscript{21,22}.

Generally, in this brightness at a scrupulous space \(d\) since the light source \(X_i\) follows the contrary square law. The glow strength of a firefly \(I\) as the space \(d\) amplifies based on \(I \propto 1/d^2\). The association of fascinated firefly \(i\) nearer a clearer firefly \(j\) can be described as follows:

\[
X_i^{t+1} = X_i^t + \beta e^{-\frac{d_i^2}{2\alpha}} (X_j^t - X_i^t) + \alpha \cdot \text{sign} (\text{rand} - \frac{1}{2}) \quad \text{Levy}
\]

where, \(X_i^{t+1}\) = modified location of firefly, \(X_i^t\) = earlier location of firefly, and \(\alpha e^{-\frac{d_i^2}{2\alpha}} (X_j^t - X_i^t)\) = attraction between fireflies.

\subsection{3.2 Bat Algorithm}

BA is created by inspiring the bio-sonar quality of microbats. BA was anticipated by mimicking the hunting potential of bats. Comprehensive examination on the BA is existing in\textsuperscript{13}.\hfill
Traditional BA (TBA) has following equations, like the velocity update, position update, and frequency vector as follows:

\[ V_i(t+1) = V_i(t) + (X_i(t) - Gbest \times F_i) \]  \hspace{1cm} (6)

\[ X_i(t+1) = X_i(t) + V_i(t+1) \]  \hspace{1cm} (7)

\[ F_i = F_{min} + (F_{max} - F_{min}) \beta \]  \hspace{1cm} (8)

where \( \beta \) is a random numeral [0,1].

### 3.3 Bacterial Foraging Optimization

Enhanced BFO is a customized form of classical BFO algorithm\(^{24,25}\). The early algorithm values are allocated as;

- Number of E.coli = \( 10 < N < 30 \) (in this work \( N = 20 \))
- \( N_x = N / 2 \);
- \( N_y = N_y \approx N / 3 \);
- \( N_{s} \approx N / 4 \);
- \( N_{r} = N / 2 \);
- \( P_{ed} = \left( N_{s} / (N + N_{s}) \right) \); \( d_{attractant} = W_{attractant} = N_s / N \); and \( h_{repellant} = W_{repellant} = N_s / N \).

### 3.4 Particle Swarm Optimization

The PSO algorithm has two essential equations like velocity and position updates, and is represented as\(^{16}\):

\[ V_i(t+1) = W \times V_i(t) + C_1 R_1(P_i - X_i(t)) + C_2 R_2(G_i - X_i(t)) \]  \hspace{1cm} (9)

\[ X_i(t+1) = X_i(t) + V_i(t+1) \]  \hspace{1cm} (10)

Where \( W \) = inertia weight (0.75), \( V_i \) = current velocity, \( V_i(t+1) \) = updated velocity, \( X_i \) = current position, \( X_i(t+1) \) = updated position, \( R_1 \) and \( R_2 \) are the random values [0,1], \( C_1 = 0.8 \) and \( C_2 = 2.2 \).

### 4. Result and Discussion

This part presents the outcome acquired with the thresholding methodology. All the simulation work is done in Matlab software on a computer with Intel core i3 CPU with 4 GB of RAM. Well known image dataset (512 x 512), such as, Mandrill, Jet, Butterfly and House is considered in this work. This work is done using the following initial algorithm limits, number of agents is selected as 20, search dimension is chosen as \( m \), stopping criteria is fixed as \( j_{max} \) and total iteration is fixed as 1000.

**Table 1.** Gray scale test images considered in this work

| Original image | Image stained with noise |
|----------------|--------------------------|
| Lena           | Gaussian                 |
| Mandrill       | Salt & Pepper            |
| Jet            |                          |
| Butterfly      |                          |
| House          |                          |

The Otsu and FA based multilevel thresholding is already applied on a class of gray scale\(^{1,2,10,15}\) and RGB images\(^{17}\) in the literature. Hence, in this paper, the robustness of the Otsu and FA based segmentation is tested using the gray scaled test images stained with the well known noise values, like the Gaussian and Salt & Pepper noises.

The default Gaussian and Salt & Pepper noises existing in Matlab is considered in this work\(^{26}\). Table 1 presents the original test images and noise stained examination images considered in this work. Table 2 presents the gray level histograms of the original and noisy images. From Table 2, one can observe that, due to the impact of the noise, the histogram levels of the test images are greatly altered, which will amplify the difficulty in the image segmentation operation. Due to the noise, the histogram distribution will be from [0, L-1]. The histogram patterns of the Butterfly and House is completely altered due to the noise.

Initially, Otsu and FA based segmentation procedure
is implemented for noise stained Lena. Table 3 shows segmented output of the Lena image for \( m=\{2,3,4,5\} \) and the output is shown in Table 3. The BA, BFO and PSO based segmentation is also applied on the above said image and the image quality are presented in Table 4. From these values, it can be observed that, FA based segmentation offers better result compared to the alternatives. From this, one can observe that, FA based approach gives expected result with considered test images compared to the alternatives. Hence, the segmentation process based on the Otsu and FA is implemented on the other test images (salt & pepper noise stained) and the results are presented in Table 5.

From this work, it can be noted that, the CPU time taken by the FA based segmentation is comparatively smaller than BA, BFO and PSO. The increase in threshold level ‘m’ will helps to achieve the improved image quality measures, such as objective function, SSIM and PSNR values.

| Table 2. Test image histograms |
|-------------------------------|

| Original histogram | Histogram of noisy image |
|--------------------|-------------------------|
| Lena               |                         |
| Mandrill           |                         |
| Jet                |                         |
| Butterfly          |                         |
| House              |                         |
Table 3. Segmented images with FA with various ‘m’

| Noise          | 2       | 3       | 4       | 5       |
|----------------|---------|---------|---------|---------|
| Gaussian       | ![Image](image1) | ![Image](image2) | ![Image](image3) | ![Image](image4) |
| Salt & Pepper  | ![Image](image5) | ![Image](image6) | ![Image](image7) | ![Image](image8) |

Table 4. Quality standards for Gaussian noise stained Lena image

| Heuristic algorithm | m | Objective function | Optimal threshold | SSIM | PSNR (dB) | CPU time (sec) |
|---------------------|---|--------------------|-------------------|------|-----------|----------------|
| FA                  | 2 | 2297.48            | 103,208           | 0.6227 | 12.4108   | 85.7152        |
|                     | 3 | 2864.30            | 72,142,224        | 0.6964 | 15.6800   | 153.6225       |
|                     | 4 | 3080.37            | 54,107,163,232    | 0.7861 | 18.0775   | 216.0459       |
|                     | 5 | 3183.29            | 41,87,129,177,240| 0.8110 | 19.9539   | 293.4856       |
| BA                  | 2 | 2831.48            | 68,138,227        | 0.6301 | 15.1406   | 153.6735       |
|                     | 3 | 3148.18            | 38,82,124,170,247| 0.7546 | 19.9228   | 293.4904       |
|                     | 4 | 3087.11            | 52,105,158,237    | 0.6738 | 18.0728   | 216.0491       |
|                     | 5 | 3180.84            | 38,84,125,172,246| 0.7399 | 19.9284   | 293.4874       |
| BFO                | 2 | 2284.73            | 98,206            | 0.6006 | 12.2085   | 85.7174        |
|                     | 3 | 2863.38            | 70,141,228        | 0.6716 | 15.5938   | 153.6227       |
|                     | 4 | 3066.25            | 51,103,157,244    | 0.6877 | 18.0085   | 216.0469       |
|                     | 5 | 3183.29            | 41,87,129,177,240| 0.8110 | 19.9539   | 293.4856       |
| PSO                | 2 | 2206.72            | 101,204           | 0.5837 | 12.2757   | 85.7208        |
|                     | 3 | 2811.56            | 66,140,218        | 0.6114 | 15.6583   | 153.6281       |
|                     | 4 | 3069.66            | 51,102,168,236    | 0.6738 | 18.0728   | 216.0508       |
|                     | 5 | 3172.84            | 39,84,126,184,247| 0.7399 | 19.9284   | 293.4874       |

Table 5. Performance measure values for Salt & Pepper noise stained image with FA

| Image  | m | Objective function | Optimal threshold | SSIM | PSNR (dB) | CPU time (sec) |
|--------|---|--------------------|-------------------|------|-----------|----------------|
| Lena   | 2 | 2349.54            | 105,214           | 0.6303 | 12.1602   | 71.5816        |
|        | 3 | 2995.37            | 77,154,224        | 0.6837 | 15.1140   | 155.8109       |
|        | 4 | 3236.45            | 55,108,169,230    | 0.7406 | 17.8500   | 248.2322       |
|        | 5 | 3326.78            | 45,87,126,177,238| 0.7991 | 19.6812   | 331.6089       |
| Mandrill | 2 | 1611.26            | 116,205           | 0.6416 | 11.0411   | 70.1314        |
|        | 3 | 2071.53            | 91,153,229        | 0.6853 | 15.2536   | 140.5982       |
|        | 4 | 2225.49            | 70,120,166,238    | 0.7196 | 17.8987   | 218.3818       |
|        | 5 | 2340.36            | 63,111,155,212,235| 0.8264 | 19.0078   | 306.6203       |
| Jet    | 2 | 2234.18            | 148,206           | 0.6390 | 11.1470   | 68.2857        |
|        | 3 | 2534.67            | 74,165,222        | 0.6927 | 14.6920   | 122.3307       |
|        | 4 | 2678.22            | 68,147,199,238    | 0.7616 | 17.9564   | 192.4319       |
|        | 5 | 2756.08            | 57,118,167,206,249| 0.7974 | 19.7322   | 278.0232       |
| Butterfly | 2 | 1667.44            | 124,197           | 0.6692 | 10.7538   | 86.2497        |
|        | 3 | 2086.18            | 95,156,208        | 0.7006 | 14.0310   | 132.1891       |
|        | 4 | 2257.84            | 73,118,169,242    | 0.7926 | 17.2900   | 226.7083       |
|        | 5 | 2355.55            | 47,113,157,214,250| 0.8118 | 18.2066   | 251.3411       |
| House  | 2 | 3417.25            | 100,188           | 0.6309 | 12.8164   | 95.9889        |
|        | 3 | 4014.52            | 64,147,204        | 0.6884 | 15.7439   | 192.5486       |
|        | 4 | 4287.36            | 46,107,179,230    | 0.7205 | 17.9465   | 258.7152       |
|        | 5 | 4407.38            | 34,79,131,193,248| 0.7608 | 19.8337   | 276.4223       |
5. Conclusion

In this manuscript, the problem of discovering optimal threshold for the 512 x 512 sized gray scale images is addressed using the Otsu’s function and Firefly Algorithm. In this paper, initial thresholds are selected as m={2,3,4,5}. To verify the sturdiness of this work, the test images are stained using the most common noise values. Cost value, SSIM, PSNR and CPU time is used to assess the eminence of segmentation procedure and then compared with the existing heuristic procedure, like PSO, BFO and BA. The simulation result confirms that; FA based segmentation helps to achieve enhanced result contrast with the alternatives for the considered noise stained images.

6. References

1. Rajinikanth V, Couceiro MS. Optimal multilevel image threshold selection using a novel objective function. Advances in Intelligent Systems and Computing. 2015; 340:177–86.

2. Ghamisi P, Couceiro MS, Benediktsson JA, Ferreira NMF. An efficient method for segmentation of images based on fractional calculus and natural selection. Expert Systems with Applications. 2012; 39:12407–17.

3. Kamalanand K, Ramakrishnan S. Effect of gadolinium concentration on segmentation of vasculature in cardiopulmonary magnetic resonance angiograms. Journal of Medical Imaging and Health Informatics. 2015; 5:147–51.

4. Manickavasagam K, Sutha S, Kamalanand K. Development of systems for classification of different plasmodium species in thin blood smear microscopic images. Journal of Advanced Microscopy Research. 2014; 9:86–92.

5. Sasirekha N, Kashwan KR. Improved segmentation of MRI brain images by denoising and contrast enhancement. Indian Journal of Science and Technology. 2015; 8:1–7. DOI: 10.17485/ijst/2015/v8i22/73050.

6. Manic KS, Priya RK, Rajinikanth V. Image multithresholding based on kapur/tsallis entropy and firefly algorithm. Indian Journal of Science and Technology. 2016; 9:1–6. DOI: 10.17485/ijst/2016/v9i12/89949.

7. Pal NR, Pal SK. A review on image segmentation techniques. Pattern Recognition. 1993; 26:1277–94.

8. Sezgin M, Sankar B. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging. 2004; 13:146–65.

9. Tuba M. Multilevel image thresholding by nature-inspired algorithms: A short review. Computer Science Journal of Moldova. 2014; 22:318–38.

10. Akay B. A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding. Applied Soft Computing. 2013; 13:3066–91.

11. Sathya PD, Kayalvizhi R. Optimum multilevel image thresholding based on Tsallis Entropy method with bacterial foraging algorithm. International Journal of Computer Science Issues. 2010; 7:336–43.

12. Sathya PD, Kayalvizhi R. PSO-based Tsallis thresholding selection procedure for image segmentation. International Journal of Computer Applications. 2010; 5:39–46.

13. Manikantan K, Arun BV, Yaradoni DKS. Optimal multilevel thresholds based on Tsallis entropy method using golden ratio particle swarm optimization for improved image segmentation. Procedia Engineering. 2012; 30:364–71.

14. Oliva D, Cuevas E, Pajares G, Zaldivar D, Perez-Cisneros M. Multilevel thresholding segmentation based on harmony search optimization. Journal of Applied Mathematics. 2013; 2013.

15. Raja NSM, Rajinikanth V, Latha K. Otsu based optimal multilevel image thresholding using firefly algorithm. Modelling and Simulation in Engineering. 2014; 2014.

16. Rajinikanth V, Raja NSM, Latha K. Optimal multilevel image thresholding: an analysis with PSO and BFO algorithms. Australian Journal of Basic and Applied Sciences. 2014; 8:443–54.

17. Rajinikanth V, Couceiro MS. RGB histogram based color image segmentation using firefly algorithm. Procedia Computer Science. 2015; 46:1449–57.

18. Rajinikanth V, Couceiro MS. Optimal multilevel image threshold selection using a novel objective function. Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing. 2015; 340:177–86.

19. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing. 2004; 13:600–12.

20. Yang XS. Firefly algorithms formultimodal optimization. Lecture Notes in Computer Science. 2009; 5792:169–78.

21. Raja NSM, Suresh Manic K, Rajinikanth V. Firefly algorithm with various randomization parameters: an analysis. Lecture Notes in Computer Science. 2013; 8297:110–21.

22. Sundaravadivu K, Sivakumar S, Hariprasad N. 2DOF PID controller design for a class of FOPTD models—an analysis with heuristic algorithms. Procedia Computer Science. 2015; 48:90–5.

23. Rajinikanth V, Aashiha JP, Atchaya A. Gray-level histogram based multilevel threshold selection with bat algorithm. International Journal of Computer Applications. 2014; 93:1–8.

24. Rajinikanth V, Latha K. Controller parameter optimization for nonlinear systems using enhanced bacteria foraging algorithm. Applied Computational Intelligence and Soft Computing. 2012; 2012.

25. Rajinikanth V, Latha K. Setpoint weighted PID controller tuning for unstable system using heuristic algorithm. Archives of Control Sciences. 2012; 22:481–505.

26. Rajinikanth V, Raja NSM, Satapathy SC. Robust color image multi-thresholding using between-class variance and cuckoo search algorithm. Advances in Intelligent Systems and Computing. 2016; 433:379–86.