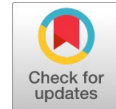


2-D histogram Based Multilevel Thresholding for Image Segmentation by Hybrid Bacterial Foraging Optimization and Particle Swarm Optimization

N S R Phanindra Kumar, P.V.G.D Prasad Reddy, G. Srinivas



Abstract: The objective of image segmentation is to extract meaningful clusters in given image. Meaningful clusters are possible with perfect threshold values which are optimized by assuming Renyi entropy as an objective function. A 1-D histogram based multilevel thresholding is computationally complex and segmented image visual quality comparatively low because of equal distribution of energy over the entire histogram plan. To overcome the problem, a 2-D histogram based multilevel thresholding is proposed in this paper by maximizing the Renyi entropy with a novel Hybrid Bacterial Foraging Optimization Algorithm and Particle Swarm Optimization (hBFOA-PSO) and the obtained results are compared with state of art optimization techniques. The results of the proposed model have been evaluated on a standard image dataset. The results obtained after implementing a 2-D histogram suggest hBFOA-PSO can be effectively used on multilevel thresholding problems resulting in a high accuracy.

Index Terms: Image segmentation; 2-D histogram, Image thresholding; Renyi entropy; Bacterial Foraging Optimization Algorithm and Particle Swarm Optimization

I. INTRODUCTION

For preprocessing of image, segmentation is an essential preliminary step. Low level and high level processing requirements are linked by these. Different varieties of segmentation are available for the application in area link recognition, detection of objects in measurements of images. The usage of segmentation is very significant for image examination such that results like success of failure are also linked with it. Reliable and accurate segmentation cannot be attained purely by automatic means. Important applications of segmentation cover detailed brain ailments as well as disease of tissue and tumour other than industrial requirements and classification of environment as in satellite, imagery optical character recognition is yet another commercially impenitent area. Divers techniques used in segmentation includes thresholding in which image are subjected to properly chosen thresholding for segmentation. Thresholding can be classified as parametric and non

parametric. In case of Ostu's entropy, the class variance is considered; other types include Tsallis's entropy, Kapur's entropy, rayni's entropy and modified Fuzzy entropy [1]. Depending on the levels, thresholding may be bi-level or multi-level. For multilevel type, multiple segments are occurs with more than two thresholds; classification can go upto six as mentioned in [2] Kapur's classification, however is based on histogram of image [3]. Another accepted method of classification is based on pixel intensity and matching class variance for dividing into regions. The above mentioned entropies are useful for bi-level thresholding and not satisfactory for the multilevel thresholding due to time consumption. Overall the chore of entropy covers, Kapur's, Birge-Shannon entropy, between class variation minimization of the Bayesian error and Masart thresholding strategy. The chief demerit of these is that computational time as well as increases as per thresholding levels. To overcome these difficulties, it is proposed to have thresholding with soft computing. bacterial foraging optimization algorithm (BFOA) has been selected for the purpose [5]. To overcome the time consumption factor, they have further modified BFOA in which the steps of swam and reproduction are made adaptive so that computational time is suddenly diminished. It is worth that alternative Active Contour Model (ACM) has been used in cuckoo search (CS) [6]. Number of soft computing options are available. For example, Bat algorithm has been utilised for maximizing Fuzzy entropy as the result have been compared with artificial bee colony algorithm (ABC), Genetic algorithm (GA) and PSO [7]. In [8], CS is based as Tsallis Entropy and the comparison is made with BF, PSO and GA's. In [9] firefly algorithm is operated on maximisation of modified Fuzzy Entropy. It is pertinent to multilevel thresholding has been tried with Ostus's entropy and Kapur's entropy; Tsallis entropy has been preferred for coloured satellite image using differential evaluation (DE) which is capable of high dimensional search space problem. In [10], such results have been compared with wind driven optimization (WDO), ABC and PSO. Very encouraging results have been obtained by Naidu et al [11] is optimizing Tsallis entropy with ACS and modified the initial results obtained through CS in which step of work required to be made adaptive for attaining global maximum. They have also tried on similar lines on firefly algorithm [12]. This paper mainly focus on getting optimal thresholds for better segmentation that is achieved with a Hybrid combination of BFOA and PSO in the category of 2-D histogram by assuming 2-D Renyi entropy as a objective/fitness function. The gained results are related with 1-D histogram and as well as with other existing algorithms.

Manuscript published on 30 August 2019.

*Correspondence Author(s)

N S R Phanindra Kumar¹, Department of Computer Science and Engineering, AITAM, Tekkali, Srikakulam., India.

P.V.G.D Prasad Reddy², Department of CS&SE, Andhra University, Visakhapatnam, India.

G. Srinivas³, Department of CSE, GIT, GITAM, Visakhapatnam, Indian.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

2-D histogram Based Multilevel Thresholding for Image Segmentation by Hybrid Bacterial Foraging Optimization and Particle Swarm Optimization

For the better understating and evaluation of presented proposed image thresholding in 2-D histogram we deliberate Structural Similarity Index (SSIM), peak signal to noise ratio (PSNR), objective/fitness function and finally Misclassification error. Several evaluations were conducted and the test results suggest that Our new approach performs better and the overall performance is high when compared with the other models such as the ACS, PSO and CS with respect to the above mentioned parameters.

In all parameters the proposed hybrid and 2-Dimensional histogram dependent image thresholding concert is better related to the supplementary algorithms.

II. CONCEPT OF RENYI ENTROPY

Let's assume an 'n' array finite discrete probability distributions (pdf) such as $(F_1, F_2, F_3, \dots, F_n) \in \Delta_n$ where $\Delta_n = \{(F_1, F_2, F_3, \dots, F_n), F_i \geq 0, i = 1, 2, 3, \dots, n, n \geq 2, \sum_{i=1}^n F_i = 1\}$ with random variables $(X_1, X_2, X_3, \dots, X_n)$ then Renyi entropy for independent and additive random events is given as [18]

$$H_\alpha = \frac{1}{1-\alpha} \log_2 (\sum_i^n F_i^\alpha) \quad (1)$$

Where ' α ' is greater than zero and it is called as entropy order. When ' α ' tends to one then Renyi entropy becomes Shannon entropy. In general image is clustered in to two clusters, one carries object information (cluster C_1) and another carries background (cluster C_2) then Renyi entropy is

$$H_\alpha[C_1] = \frac{1}{1-\alpha} \left[\log_2 \left(\sum_{i=0}^t \left(\frac{F(i)}{F(C_1)} \right)^\alpha \right) \right] \quad (2)$$

$$H_\alpha[C_2] = \frac{1}{1-\alpha} \left[\log_2 \left(\sum_{i=t}^{L-1} \left(\frac{F(i)}{F(C_2)} \right)^\alpha \right) \right] \quad (3)$$

Where $F(C_1) = \sum_{i=0}^t F(i)$, $F(C_2) = \sum_{i=t+1}^{L-1} F(i)$, Here F_i is the normalized one dimensional histogram of the image and ' L ' is highest intensity level of gray scale image. The overall renyi entropy for a given image with one threshold ' t ' is given as

$$\varphi_\alpha(t) = \argmax([H_\alpha[C_1] + H_\alpha[C_2]]) \quad (4)$$

2.1 Multi-Level Thresholding: Let image is divided into ' N ' number of clusters $C = (C_1, C_2, C_3, \dots, C_N)$ with N number of threshold values $t = (t_1, t_2, t_3, \dots, t_N)$ then Renyi entropy for each individual cluster is defined as [13]

$$H_\alpha[C_1] = \frac{1}{1-\alpha} \left[\log_2 \left(\sum_{i=0}^{t_1} \left(\frac{F(i)}{F(C_1)} \right)^\alpha \right) \right] \quad (5)$$

$$H_\alpha[C_2] = \frac{1}{1-\alpha} \left[\log_2 \left(\sum_{i=t_1+1}^{t_2} \left(\frac{F(i)}{F(C_2)} \right)^\alpha \right) \right] \quad (6)$$

$$H_\alpha[C_N] = \frac{1}{1-\alpha} \left[\log_2 \left(\sum_{i=t_{N-1}}^{L-1} \left(\frac{F(i)}{F(C_N)} \right)^\alpha \right) \right] \quad (7)$$

Where $F(C_1) = \sum_{i=0}^{t_1} F(i)$, $F(C_2) = \sum_{i=t_1+1}^{t_2} F(i)$, $F(C_N) = \sum_{i=t_{N-1}}^{L-1} F(i)$, the overall Renyi entropy or objection function for a given image for N thresholds is given as $\varphi_\alpha(t) = \argmax([H_\alpha[C_1] + H_\alpha[C_2] + \dots + H_\alpha[C_N]])$ (8)

For simplifying the calculations, two dummy thresholds are introduced t_0 and $t_N = L-1$ which satisfy the condition $t_0 < t_1, \dots, t_{N-1} < t_N$. The optimal thresholds are obtained by maximizing the above equation with any soft computing technique.

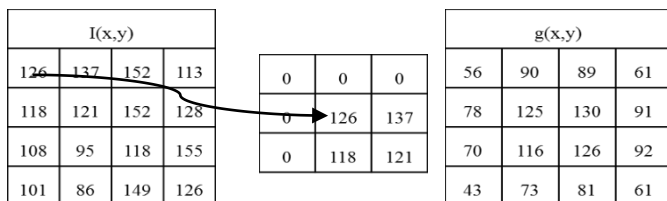


Figure 1. Example for 2-D histogram calculation

Retrieval Number: J98340881019/19©BEIESP

DOI: 10.35940/ijitee.J9834.0881019

Journal Website: www.ijitee.org

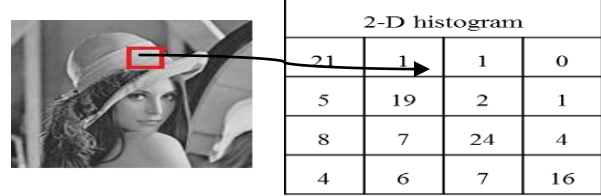


Figure 2. Lena image and 2-D histogram

Two-Dimensional Renyi Entropy:

Let $I(m,n)$ is an image intensity at spatial location (m,n) . In digital image $[I(m,n) | m \in \{1, 2, 3, \dots, M\}, n \in \{1, 2, 3, \dots, N\}]$, where ' M ' and ' N ' are size of the image and its 1D-histogram $h(x)$ for $x \in \{1, 2, 3, \dots, L-1\}$, where ' L ' is 256 for gray scale image. Let denote elements in histogram $\{1, 2, 3, \dots, 255\}$ as G . In literature, optimal thresholds selection is based on 1D-histogram and is obtained by optimizing the objective function/entropy.

The 2-D histogram of an image is obtained by defining a local average of pixel, $I(x,y)$, as the average intensity of its nine neighbors denoted as $g(x,y)$ [14]

$$g(x,y) = \frac{1}{9} \sum_{i=-1}^1 \sum_{j=-1}^1 f(x+i, y+j) \quad (9)$$

For example let us take an image of size 4×4 as shown in figure 1 (a) and its average intensity $g(x,y)$ is calculated by padding required number of zeros at edges as shown in figure 1 (b). First table in figure is image and first element i.e 126, $g(x,y)$ is calculated by padding zero's at edges as in figure and last tables shows $g(x,y)$ for entire image $I(x,y)$. 2-D histogram of Lena image at marked area is shown in figure 2, where diagonal quadrants carry much information.

The 2-D histogram of tested images as shown in figure 2 and are divided into four clusters by a single threshold (t,s) . Where t is threshold for original image intensity $I(x,y)$ and s is threshold for average intensity image $g(x,y)$. The divided cluster area is not same. The diagonal quadrant 1st represents object and 3rd represent background and 2nd and 4th quadrants are neglected because does not carry any information (pair occurrence is less) as show in figure. The Renyi entropy for object and background is given as

$$H_{object}^\alpha[t,s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=0}^t \left(\sum_{j=0}^s \left(\frac{F(i,j)}{F_{D1}(t,s)} \right)^\alpha \right) \right] \quad (10)$$

$$H_{background}^\alpha[t,s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=t+1}^{L-1} \left(\sum_{j=s+1}^{L-1} \left(\frac{F(i,j)}{F_{D2}(t,s)} \right)^\alpha \right) \right] \quad (11)$$

Where $F_{D1}(t,s) = 1 - \sum_{i=0}^t \sum_{j=0}^s F(i,j)$ and $F_{D2}(t,s) = 1 - \sum_{i=t+1}^{L-1} \sum_{j=s+1}^{L-1} F(i,j)$

The final objective function which is to be maximized for better threshold (t,s) selection is

$$\varphi_\alpha(t) = \argmax([H_{object}^\alpha[t,s] + H_{background}^\alpha[t,s]]) \quad (12)$$

2.2 Proposed Renyi 2d-Histogram Based Multi-Level Thresholding

Multilevel thresholding with 1-D histogram deliver inferior results because of incorrect selection of thresholds, so recent study proved that thresholding with 2-D histogram deliver superior results especially in multilevel thresholding. Multilevel thresholding gained lots of popularity over bi-level thresholding because, it clusters the image into several useful clusters, helps for accurate analysis and interpretation of the image.



Published By:

Blue Eyes Intelligence Engineering

and Sciences Publication (BEIESP)

© Copyright: All rights reserved.

In this paper, we proposed a 2-D Renyi entropy based multilevel thresholding for image segmentation by incorporating the advantage of 2-D histogram. If the 2-D histogram of an image is cluster into 9 clusters with two thresholds (t_1, t_2) and (s_1, s_2) as shown in figure 3 (a). Then the diagonal quadrants 1st, 5th and 9th represents objects(s), intermediate regions and background respectively as illustrated in Figure 3 (a) and reset of the regions are noise and edges and are ignored. The Renyi entropy of diagonal quadrants are given as

$$H_{object}^{\alpha}[t, s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=0}^{t_1} \left(\sum_{j=0}^{s_1} \left(\frac{F(i, j)}{F_{D1}(t, s)} \right)^{\alpha} \right) \right] \quad (13)$$

$$H_{intermediate}^{\alpha}[t, s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=t_1+1}^{t_2} \left(\sum_{j=s_1+1}^{s_2} \left(\frac{F(i, j)}{F_{D2}(t, s)} \right)^{\alpha} \right) \right] \quad (14)$$

$$H_{background}^{\alpha}[t, s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=t_2+1}^{L-1} \left(\sum_{j=s_2+1}^{L-1} \left(\frac{F(i, j)}{F_{D3}(t, s)} \right)^{\alpha} \right) \right] \quad (15)$$

Where $F_{D1}(t, s) = 1 - \sum_{i=0}^{t_1} \sum_{j=0}^{s_1} F(i, j)$

$F_{D2}(t, s) = 1 - \sum_{i=t_1+1}^{t_2} \sum_{j=s_1+1}^{s_2} F(i, j)$ and

$$F_{D3}(t, s) = 1 - \sum_{i=t_2+1}^{L-1} \sum_{j=s_2+1}^{L-1} F(i, j)$$

The final objective function which is to maximized for better threshold (t, s) selection is

$$\varphi_{\alpha}(t) = \operatorname{argmax}([H_{object}^{\alpha}[t, s] + H_{intermediate}^{\alpha}[t, s] + H_{background}^{\alpha}[t, s]]) \quad (16)$$

Above equation can be extended for 'N' threshold values as given below

$$\varphi_{\alpha}(t) = \operatorname{argmax}([H_1^{\alpha}[t, s] + H_2^{\alpha}[t, s] + H_3^{\alpha}[t, s] \dots \dots H_{N+1}^{\alpha}[t, s]) \quad (17)$$

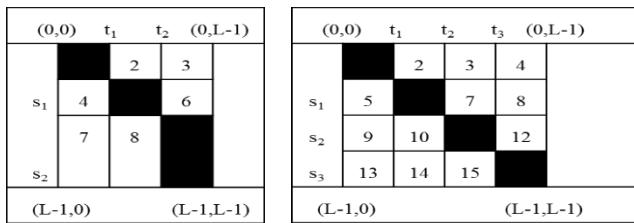


Figure. 3. 2-D histogram for a) 3- level b) 4- level.

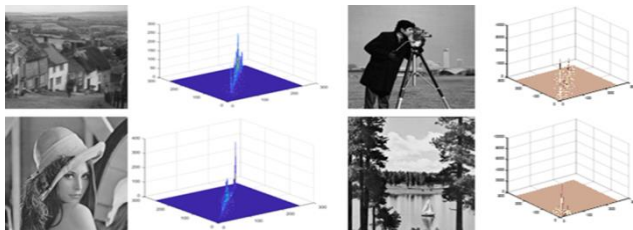


Figure. 4. Input images and corresponding 2-D histogram Where

$$H_k^{\alpha}[t, s] = \frac{1}{1-\alpha} \left[\log_2 \sum_{i=t_{k-1}+1}^{t_k} \left(\sum_{j=s_{k-1}+1}^{s_k} \left(\frac{F(i, j)}{F_{Dk}(t, s)} \right)^{\alpha} \right) \right] \quad (18)$$

For simplifying the calculations, two dummy thresholds are introduced t_0 and $t_{N+1} = L-1$ which satisfy the condition $t_0 < t_1 \dots \dots < t_{N-1} < t_{N+1}$. Similarly two dummy variable s_0 and $s_{N+1} = L-1$ which satisfy the condition $s_0 < s_1 \dots \dots < s_{N-1} < s_{N+1}$. The 2-D histogram of four standard images are shown in figure 4 and form these figure it is observed that most of the information/energy is concentrated on diagonal quadrants. Multilevel thresholding is a time consuming process and is proportional to the number of thresholds 'N'. So soft computing techniques play a significant role in this contest by assuming eqn. (17) as an objection function, which leads to reduction in the computational time.

III. PROPOSED hBFOA-PSO

This paper mainly focuses on optimization of thresholds levels for optimal image segmentation. An attempt is made on the basis of hybridizing the two well-known metharustic optimization techniques such as particle swam optimization (PSO) and bacterial foraging optimization algorithm (BFOA). The BFOA being a global search algorithm it found many applications in the field of engineering and medical applications and sometimes it may follow into local optimal solution. To avoid the limitations of BFOA, a PSO algorithm is introduced which speedup the execution time and avoids being follow in local optimal solution. The hybridized algorithm is called hBFOA-PSO which is further used for optimal thresholding for effective image segmentation and obtained evaluations are compared with state of art optimization algorithms. In the following section all the algorithms are explained clearly in all accepts.

3.1 Bacteria Foraging Optimization Technique: Overview

The scientist named Passino introduced BFOA in 2002 and is algorithm is developed based on in depth study on behaviour of Bacteria in process of foraging for food in any substances [15]. The bacteria named *E.Coli* gains energy by searching for nutrients per every minute. This nutrients search may be occurred by sharing information among the bacteria's. Another way of getting nutrients is by following three steps: swarming or tumbling and chemotaxis step. In algorithm structure of BFOA, search process is extraordinary because of its inbuilt well established algorithm follow. In searching process of food, BFOA follows a step size based on Gaussian distribution function, whereas cuckoo search following Levydistribution. The optimal thresholds are acquired by optimizing the Renyi's entropy in two dimensional with BFOA. The algorithm directs all bacteria's towards maximizing the Renyi's entropy fitness function. Initially all the bacteria's are randomly selected so each one carries different objective values and in every iterations these values are updated by learning themselves. The highest objective function holding bacteria is forwarded to next iterations and least bacteria are replaced with new ones. In each iterations rest all bacteria's try to follow the highest objective bacteria. This process is repeated until the required objective function value achieved. The BFOA achieve this optimal result in four poured stages: 1. Chemotaxis, 2.Reproduction 3.Elimination-dispersal 4.swarming. These four stages of BFOA are explained below.

1. Chemotaxis: This stage is a critical stage in BFOA in incisive for food, Chemotaxis is exemplifies astuteness functional by the bacteria in incisive for food. The bacteria move towards the healthier result by attractive step by swimming or thumbling. In BFOA algorithm, every bacteria moves to healthier position by derivative of eight neighborhoods positions. After this derivative, all bacteria's finds maximum which bacteria holding highest objective function value and rest bacteria's track the bacteria of holding maximum objective function value. The steps in Chemotaxis:

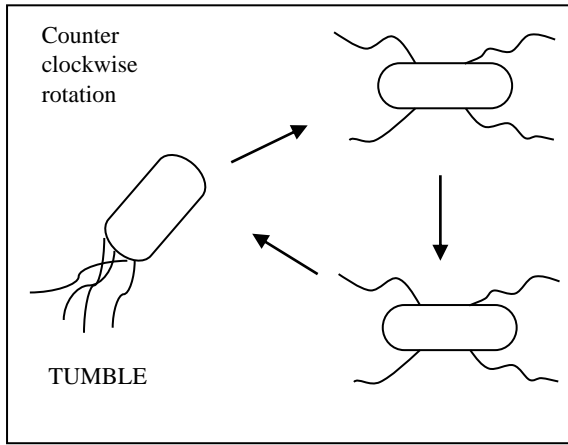


Figure 5. Bacterium tumbling stage

Tumbling: In this stage bacteria move arbitrarily in a certain direction where extraordinary nutrients are existing in search space. All the bacteria have initially some nutrients naturally. This procedure is identified as tumbling process as shown in Figure 5.

Swimming (up): Nutrient obtained after tumbling, if those are sufficient and successful then bacteria move in same path in order to increase further or else it go for swimming process. This movement of swimming is called swimming up.

Swimming (down): If bacteria moving path decrease the bacteria nutrients then that movement are known as swimming down. When bacteria finds swimming down stage then instantly bacteria change its path. Figure 6 shows swimming stage.

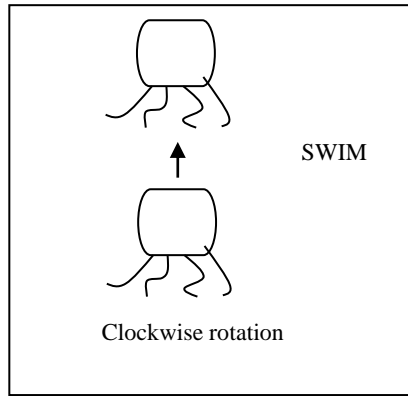


Figure 6. Swimming stage bacteria

The bacteria step of chemo tactic is given in Eq. (19).

$$\theta^x(p+1, q, r) = \theta^x(p, q, r) + C(x) \cdot \frac{\Delta(x)}{\sqrt{\Delta^i(x)\Delta(x)}} \quad (19)$$

Where $C(x)$ is bacteria step size and $\Delta(x)$ is randomly chosen floating number between $[0,1]$.

2. Swarming: In this stage bacterium tries to communicate with each other for better improve of their positions towards getting better and high quality nutrients for better survival or to extend their life span. So remaining bacteria always try to moves toward highest nutrients bacteria direction and circumvents the path of movement toward the lowest nutrient. This process usually called swarming and below equation described this step.

$$\begin{aligned} J_{cc}(\theta, (P(p, q, r))) &= \sum_{i=1}^S J_{cc}(\theta, \theta^i(P(p, q, r))) \\ &= \sum_{x=1}^S \left[-d_{att} \exp \left(-W_{att} \sum_{n=1}^m (\theta_n - \theta_n^x)^2 \right) \right] \\ &+ \sum_{x=1}^S \left[-h_{rep} \exp \left(-W_{rep} \sum_{n=1}^m (\theta_n - \theta_n^x)^2 \right) \right] \end{aligned} \quad (20)$$

3. Reproduction: Before applying this step on bacteria, all the bacteria are assigned ranked based on fitness value either in descending or ascending order depends on maximization or minimization problem respectively. As entropy must to be maximum so in this paper all the bacteria's are ordered in ascending order as per their objective values. In this stage half of the bacteria's with lowest ranks are died and are replaced with the new generation generated by mutation between two largest objective value bacteria's. So that number of populations or solutions in search space is same. This entire process is called conjugation.

4. Elimination-dispersal: In some situations bacteria may familiarity an unexpected change in eco friendly situations like hike in humidity or in temperature. Then bacteria experiences third stage i.e. reproduction, in which bacteria dies because of unexpected changes and new bacteria's are produced by a sexual relative between two other bacteria. Some bacteria's may be moved to the nearest and safest place.

BFOA Algorithm:

For $i = 1:M_{ed}$

For $j = 1:M_{re}$

For $k = 1:M_c$

For $l = 1:SU$

$J(i, j, k, l) = J(i, j, k, l) + J_{cc}[\theta^p(i, j, k), P(i, j, k)] \Delta_m(p), m = 1, 2, \dots, p$

$$\theta^i(q+1, r, s) = \theta^i(i, j, k) + C(i) \cdot \frac{\Delta(i)}{\sqrt{\Delta^i(i)\Delta(i)}}$$

Calculate the $J(i, j+1, k, l)$

$p = 0;$

While $p < Ml$

$MM = M+1$

If $J(i, j, k, l) < J_{last1}$

update J_{last1}

$$\theta^i(q+1, r, s) = \theta^i(i, j, k) + C(i) \cdot \frac{\Delta(i)}{\sqrt{\Delta^i(i)\Delta(i)}}$$

else $mmm = M_s$

End if

End while

End j

End k

End l

For $i = 1:SS$

$$J_{helth} = \sum_{j=1}^{NNc+1} J(i, j, k, l)$$

```

End i
End j
For i = 1:SSS
If randn () < Pedd
Unhealthy bacteria will be eliminated and swap with a new
randomly produced
End if
End p
End l

```

3.2 Particle Swarm Optimization (PSO)

Kennedy and Eberhart proposed a new optimization i.e PSO in 1995. The PSO is a stochastic methodology under swarm based intelligence and it mimics in what way particle is moving to get best food for survival [16]. Every particle individually and adaptively updates the position and velocity around the search space based on the experience of previously obtained knowledge around its space and the understandings of additional particle in the populations. Every particle is given with a temporary memory by which particle store the good food location it forever visited during its expedition. Particle good food location is treated as Pbest and the best good location of group occupied as one is stowed as Gbest. The preliminary positions of Pbest and Gbest are dissimilar. Particle is showed to give the better results in procurement the global maxima or minima. However, getting the global minima/maxima optimum value is a stimulating matter, whenever there will be multiple minima happens. This PSO algorithm doesn't involve mutation or crossover operators. It depends on initialization of the tuning parameters, the swarm size, the objective/fitness function and the minima/maxima number of iteration. It doesn't depend on the preliminary conditions and the incline values.

The compensations of using the PSO are less expensive in computationally, abundant simple to instrument, Lower CPU time and requirement of memory[19]. The each particle modified velocity is given in Eq (21)

$$velocity(t+1) = velocity(t) + c_1 r_1 (Pbest - x(t)) + c_2 r_2 (Gbest - x(t)) \quad (21)$$

The each particle position modified with below equation
 $xpos(t+1) = xpos(t) + velocity(t+1)$ (22)

Algorithm of PSO:

Step 1: Initialize every individual particle in solutions with randomly position & randomly velocity.

Step 2: Calculate the objective/cost function of every particle. If the present objective/cost is sophisticated than the finest value up to now calculated, then it is stowed in Pbest.

Step 3: Find the highest objective/cost particles among all and assume that position is Gbest.

Step 4: Find the fresh velocity and position of every particle rendering to the above equations. (1, r, s)

Step 5: Reiteration the above all steps from 2-4 till maximum iterations or minimum/maximum criteria.

3.3 hBFOA-PSO ALGORITHM

The hBFOA-PSO syndicates PSO and BFOA algorithms, so it earnings the disadvantages and advantages of both algorithms. The goal is to share/support information among BFOA and PSO that pointers to generation of healthy and wealthy bacteria by means of elimination and dispersal. The main disadvantage in BFOA is, tumbling step is random in all iterations, so attaining a global solution/result is difficult. Whereas, in proposed hybrid BFOA-PSO algorithm, the tumbling step is not random in every iterations and with the help of PSO these steps of tumbling is

optimized. The suitable & better tumbling step and global best result from PSO is prearranged as input to the BFOA. Tumbling steps are updated at first step of BFOA. The parameter notations for hybrid BFOA and PSO given is below.

Step 1: parameters Initialization for both PSO and BFOA:

pp = Problem dimensions;

SS = population/solutions size or in case of PSO number of particle and number of bacteria in case of BFOA;

NN_s = length of swim in chemotaxis loop, followed by tumbling stage;

NN_c = stopping criteria or iterations maximum

NN_{re} = Max reproduction step;

NN_{ed} = Max number of steps in case of elimination and dispersal;

PP_{ed} = probability of dispersal and elimination

CC(i) = tumbling step size; d_{at} , w_{at} , h_{re} , w_{re} = Bacteria repellent and attractive coefficients;

Δ (pp, ii) = Bacteria's direction in present iteration;

P (ii, jj) = Bacteria's position in present iteration;

c_1 , c_2 = PSO tuning parameters;

r1, r2 = randomly selected numbers [0 to 1] in PSO;

Step 2: dispersal and Elimination loop: ss = ss + 1.

Step 3: Reproduction loop: rr = rr + 1.

Step 4: Chemotaxis loop: qq = qq + 1.

Sub step aa: For pp = 1, 2, . . . , SS, ith bacteria's moves with following steps

- Find all bacteria's objective/fitness value, $J_j(i,j,k,l)$; henceforth new fitness/objective function is $J_j(i,j,k,l) = -J_j(i,j,k,l) + J_{ccc}(hh(j,k,l), Pp(j,k,l))$;

- Assign $J_{j_last} = J_j(j,k,l)$.

Sub step bb: For pp = 1, 2, . . . , SS bacteria's categorical to take either swimming or tumbling $\Delta(ii)$, that is random generation lies numbers 0 to 1 between in first iterations for all bacteria's or for all solutions/populations. From second onwards iterations tumbling and directions are optimized with the help of PSO. Bacteria's move towards best direction with $\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i) \cdot \frac{\Delta(i)}{\sqrt{\Delta^i(i)\Delta(i)}}$

Where j, k and l are indexes of Chemotaxis steps, re-production and Elimination & dispersal respectively.

Which leads pth bacteria will move with a step size of Cc(i) in tumbling stage

Calculate $J(i,j,k,l) = J(i,j,k,l) + J_{ccc}[\theta^p(j, k, l), P(j, k, l)]$

- Stage: Swimming

- i. Assume mm = 0 (hostage for swim length).

- ii. While mmm < NN_s

Let mmm = mmm + 1

If $J(i,j+1,k,l) < J_{j_last}$, let $J_{j_last} = J(i,j+1,k,l)$ and $\theta^i(j+1, k, l) = \theta^p(j, k, l) + C(i) \cdot \frac{\Delta(i)}{\sqrt{\Delta^i(i)\Delta(i)}}$

Sub-step c: If pp ≠ SSs Next bacteria (j + 1) (i.e., go to sub-step b for next bacteria)

Step 5: calculate global and local best positions of each bacteria's.

Step 6: updated every bacteria's velocity and position with PSO. New update vector $\Delta(pp, p)$.

Step 7: If j < NN_c, go to stage 4 and repeats chemotxis step until the bacteria's alive.

Step 8: Reproduction loop:



2-D histogram Based Multilevel Thresholding for Image Segmentation by Hybrid Bacterial Foraging Optimization and Particle Swarm Optimization

- Sub-step a: Find fitness of bacterium p by significant the s and r value, j_{HL}^i as $j_{HL}^p = \sum_{j=1}^{N_c+1} j(i, j, k, l)$. j_{HL}^p show the amount of nutrients a bacteria has over its total generation and it's a quantity of success of bacterium in circumventing noxious materials. As per the j_{HL} values, Category all bacterium in rising order and also chemotactic parameter C (p).
- Sub-step bb: The $SS_r = SSS/2$, the uppermost j_{HL} values of bacteria are uninvolved and rest S_r bacterium with the finest j_{HL} value divided. New bacteria are placed at the same location.

Step 9: If $ss < NN_{re}$, moves to step 3.

Step 10: dispersal and Elimination: For $j = 1, 2, \dots, SSS$, with probability Pp_{ed} , eliminates and disperses each bacteria.

IV. RESULTS AND DISCUSSION

The proposed algorithm performance is evaluated by considering the standard benchmark images like Lena, Goldhill, Lake and pirate and all the images are of size 256×256 and each pixels take 8 bits (bits per pixel = 8). All the images are in '.tif' format except Lake which is in '.gif' and Goldhill is in '.jpeg' format as shown figure 7. All the algorithms are simulated in Mat lab version 2017 and implemented in desktop with specifications: Windows 7 Enterprise N, HP Compaq LE1902x, Intel (R) Core (TM) Duo CPU e7500N at 2.93GHz, 64-bit operating system. The number of iterations $itr = 50$, population size $P = 100$, upper bound $U_b = 255$, $L_b = 0$, dimensions of the problem $D = th$ are consider for all optimization algorithms. In this paper, number of thresholds $t = 5$ selected for all algorithms because the number of thresholds in published previous paper is 5. The proposed approach is tested by thresholding the image with the help of both 1-D histogram and 2-D histogram and compared the results with the ACS, PSO and CS. The same tuning parameters as in [17] are taken for CS and PSO.

Table 1: Comparison of Objective Value & Standard deviation for various algorithms.

Images	Opt Tech	Objective value		Standard deviation	
		1-D histogram	2-D histogram	1-D histogram	2-D histogram
Lake	PSO	10.120562	12.47854	0.143086	0.245784
	CS	10.347841	12.65987	0.257786	0.457895
	ACS	10.569789	12.78548	0.578977	0.245789
	hBFOA-PSO	10.654877	12.89787	0.145789	0.258745
Lena	PSO	14.193136	16.25879	0.137058	0.247845
	CS	14.258797	16.35987	0.245784	0.549777
	ACS	14.358974	16.58798	0.247895	0.258789
	hBFOA-PSO	14.457895	16.69696	0.247854	0.247841
Pirate	PSO	13.749391	15.01478	0.091611	0.147854
	CS	13.897544	15.25874	0.014785	0.258745
	ACS	14.025844	15.45789	0.587945	0.578454
	hBFOA-PSO	14.258745	15.65877	0.258794	0.258744
Goldhill	PSO	13.965610	15.02589	0.094859	0.001477
	CS	14.012471	15.25789	0.258794	0.002145
	ACS	14.258974	15.36985	0.258741	0.014578
	hBFOA-PSO	14.478954	15.58974	0.236589	0.025874

4.1. Quantitative validation

To inspect the influence of the hBFOA-PSO algorithm for the problem of multilevel thresholding, we considered Renyi entropy as objective function or fitness function. The Hybrid Bacterial Foraging Optimization Algorithm and particle swarm optimization and other three algorithms are applied on Renyi entropy objective function and the results of the hBFOA-PSO are compared with ACS, PSO and CS in both 1-D and 2-D histogram. To maximize the objective function all the algorithms are optimized. Table 1 shows the objective function for hBFOA-PSO, ACS, PSO and CS. Hence from Table. 1 by using Renyi entropy the objective value obtained with 2-D histogram for different images are higher than with 1-D histogram and proposed

hBFOA-PSO objective value is higher than ACS, PSO and CS with both histograms.

4.2 Peak Signal to Noise Ratio and Mean Square Error

The PSNR illustrates the variations among the threshold image and input image. In general the measure of visual difference of two images and units are decibels (dB). If the reconstructed image shows the better quality, it indicates the higher value of PSNR.

The below equation (23) to calculate Peak signal to noise ratio value, f_x is output vector and f_y is input vector shown below.

$$PSNR = 10 * \log_{10} \left(\frac{(255^2)}{MSE} \right) \quad (23)$$

The below equation (24) to calculate Mean square Error value, Y is output image and X is input image and $M \times N$ is the size of image shown below.

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (x_{ij} - y_{ij})^2 \quad (24)$$

The values attained for the PSNR from the different algorithms are shown in Table 2, when compared to ACS, PSO and CS, the proposed algorithm attains higher PSNR value with 2-D histogram as compared to 1-D histogram. So the quality of the reconstructed images gets much better for the higher level of thresholds. Also, results displayed in table 2, suggest that A 2-D Histogram on the image set produce a much better PSNR than the 1-D histogram and quality of images reconstructed from the 2-D histogram fare much better than its 1-D counterpart. The mean square error between the reconstructed and original image is less for hBFOA-PSO compared to other methods as tabulated in Table 2.

4.3. Misclassification error:

It shows the segment numbers which are misclassified in between the original images and segmented images. Assume one pixel actually is in foreground but unfortunately treated as background pixel then we treat that pixel is misclassified. In similarly, a collection of pixels comes underneath background but those are treated as foreground then the resultant segmentation is misclassified and is calculated by below Eq. 25

$$M = 1 - 2 * t * \frac{\sum_{j=0}^{Th} \sum_{i \in R_j} (I_i - \sigma_j)^2}{N * (I_{max} - I_{min})^2} \quad (25)$$

Table 2: Comparison of PSNR & MSE values for various algorithms.

Images	Opt Tech	PSNR		MSE	
		1-D histogram	2-D histogram	1-D histogram	2-D histogram
Lake	PSO	32.8586	33.2654	33.66824	30.65776
	CS	32.9874	33.3589	32.68439	30.00477
	ACS	33.1478	33.4789	31.49926	29.18706
	hBFOA-PSO	33.2589	33.5987	30.70368	28.39294
Lena	PSO	28.6039	29.3698	89.67892	75.17966
	CS	28.7894	29.4578	85.92912	73.67164
	ACS	28.8974	29.5897	83.81859	71.46779
	hBFOA-PSO	28.9874	29.6589	82.09947	70.33806
Pirate	PSO	27.7229	28.1457	109.8475	99.65759
	CS	27.8974	28.2589	105.5214	97.09355
	ACS	27.9874	28.4785	103.3571	92.30609
	hBFOA-PSO	28.0145	28.5874	102.7142	90.02028
Goldhill	PSO	29.0785	30.1589	80.39524	62.68886
	CS	29.1457	30.2589	79.16083	61.26189
	ACS	29.2587	30.3589	77.1277	59.86739
	hBFOA-PSO	29.3698	30.5879	75.17966	56.79242

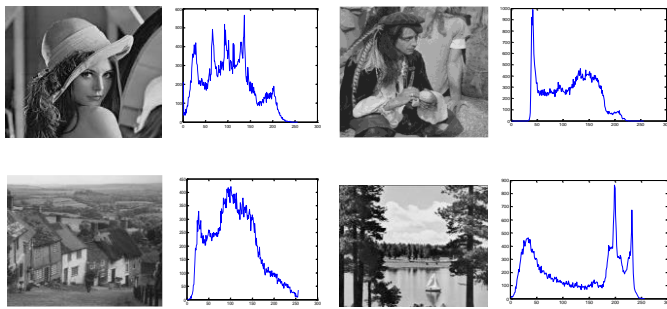


Fig.7. tested images and corresponding histograms a) Lena b) Pirate c) Goldhill d) Lake

Where t is number of required segment regions or dimensions of thresholds, R_{jts} is j^{th} segment regions, pixel intensity levels are I_{at} at definite segment region, σ_{jts} j^{th} mean of all segment regions, image size is N (number of rows * column), I_{min} is lower intensity of input image (between 0 in case of gray scale image and three for color) and I_{max} higher intensity level of an image (255 in case of gray scale). Many times, misclassification error does lay in-between 0 to 1 and lower value of misclassification value shows to best image segmentation and highest misclassification shows nastiest segmentation. Henceforth, the misclassification error, in segmentation is measure of difference in-between maximum 1 (better quality in visual) and minimum i.e 0 (not good quality in image). A comparison of the hBFOA-PSO with respect to the other models is made and the Misclassification error and SSI are displayed in Table. 3. The results ascertain that our proposed model performs better with a lowest error and better segmentation is achieved when used in combination with a 2-D histogram than 1-D histogram.

Table 3: Misclassification error & Structural Similarity Index

Images	Opt Tech	Misclassification error		Structural Similarity Index	
		1-D histogram	2-D histogram	1-D histogram	2-D histogram
Lake	PSO	0.582330	0.514785	0.674714	0.685954
	CS	0.578797	0.501245	0.685954	0.695844
	ACS	0.568874	0.497851	0.698542	0.701254
	hBFOA-PSO	0.558797	0.485125	0.699999	0.714587
Lena	PSO	0.622223	0.595454	0.560888	0.584561
	CS	0.614785	0.586412	0.578954	0.598745
	ACS	0.601258	0.578954	0.589451	0.601425
	hBFOA-PSO	0.596547	0.569852	0.598741	0.614521
Pirate	PSO	0.694293	0.678954	0.522832	0.535895
	CS	0.689541	0.665985	0.536985	0.548721
	ACS	0.674582	0.658974	0.548751	0.558741
	hBFOA-PSO	0.665474	0.645874	0.558962	0.569842
Goldhill	PSO	0.617677	0.632589	0.603996	0.625874
	CS	0.602589	0.625874	0.614584	0.632594
	ACS	0.589541	0.614575	0.625896	0.647854
	hBFOA-PSO	0.575412	0.602589	0.635421	0.658974

4.4. Structural Similarity Index (SSIM)

It guesses the visual resemblance in between input and segmented/threshold image and is measured from Eq. (26)

$$SSIM = \frac{(2\mu_I\mu_{\tilde{I}} + C1)(2\sigma_{\tilde{I}} + C2)}{(\mu_I^2 + \mu_{\tilde{I}}^2 - C1)(\sigma_I^2 + \sigma_{\tilde{I}}^2 - C2)} \quad (26)$$

Where μ_I and $\mu_{\tilde{I}}$ are the mean of input I and segmented \tilde{I} image, σ_I and $\sigma_{\tilde{I}}$ are the standard deviation of input I and segmented \tilde{I} , $\sigma_{\tilde{I}}$ is cross correlation and $C1$ and $C2$ are fixed values and is 0.065 in this work. Table. 3 demonstrate the SSIM for various techniques along Renyi entropy and it reveal proposed higher SSIM with 2-D histogram as compared to 1-D histogram and is higher than other methods too.

4.5. Qualitative Results

Here we focused on visual clarity of reconstructed image which are obtained by thresholding the image in both 1-D histogram and proposed 2-D histogram by maximizing the Renyi entropy with proposed hybrid BFOA-PSO and with ACS, PSO and CS. The segmented images and histogram

obtained with 1-D histogram and proposed 2-D histogram with hybrid BFOA-PSO algorithms at threshold level 5 with Renyi entropy are shown in Figure 8. As we know, at higher levels of threshold (i.e, $t = 5$) compared to $t = 2$, $t = 3$ and $t = 4$ the constructed image visual quality is much better. For efficiency measure of proposed Hybrid BFOA-PSO, let us have a look at the visual quality of few segmented images with Renyi entropy: Pirate image at 5 level thresholds as in Figure. 8b, Lena image at 5 level thresholds as in Fig. 7a from these figures h BFOA-PSO segmented visual quality is better with 2-D histogram as compare to 1-D histogram. Similarly, for Renyi entropy segmented visual quality of proposed hBFOA-PSO is better with 2-D histogram. For example, Goldhill image at 5 level thresholds as shown in Fig.8c and Lake image at 5- level thresholds as shown in Fig.8d. When comparing the quality of an image, the proposed algorithm performs better than the existing approaches. In Fig. 8d, Visibility of the background lake is poor with 1-D histogram at five level thresholds. But it is clearly visible with 2-D histogram. Moreover as the number of thresholds is increased the images becomes clearly recognizable.



Figure 8. Segmented images with 1-D and 2-D histogram respectively a) Lena b) Pirate c) Goldhill d) Lake

V. CONCLUSIONS

To maximize the Renyi entropy, a Hybrid Bacterial Foraging optimization algorithm in combination with PSO is used on a 2-D Histogram. This is approach is based on a multilevel threshold technique for image segmentation. To evaluate the performance of this proposed approach, the algorithm is implemented on a standard image dataset. The procured result of the hBFOA-PSO is evaluated against the other algorithms such as CS, PSO and ACS with Renyi entropy. With these comparisons, it is observed that the proposed algorithm i.e., hBFOA-PSO has a maximum fitness value among other algorithms. The proposed algorithm has higher PSNR, objective value and SSIM values and lower misclassification error and mean square error than CS, PSO and ACS and more over improved segmentation image quality is obtained with proposed method. It is concluded that the proposed algorithm outperforms CS, PSO and ACS in all performance parameters with 2-D histogram and 1-D histogram.

2-D histogram Based Multilevel Thresholding for Image Segmentation by Hybrid Bacterial Foraging Optimization and Particle Swarm Optimization

REFERENCES

1. De Luca, A. S. Termini, A definition of non-probabilistic entropy in the setting of fuzzy sets theory, *Inf. Control* 20 (1972) 301–312.
2. Sezgin, M. B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, *J. Electron. Imaging* 13 (1) (2004) 146–165.
3. Kapur, J. N, P.K.Sahoo, A.K.C Wong, A new method for gray-level picture thresholding using the entropy of the histogram”, *Computer Vision Graphics Image Process.* 29 (1985) 273 – 285.
4. Otsu, N, “A threshold selection from gray level histograms” *IEEE Transactions on System, Man and Cybernetics* 66, 1979.
5. Sathya. P. D and R. Kayalvizhi, “Optimal multilevel thresholding using bacterial foraging algorithm”, *Expert Systems with Applications*, Vol. 38, pp. 15549–15564, 2011.
6. Mbuyamba, M, J. Cruz-Duarte, J. Avina-Cervantes, C. Correa-Cely, D. Lindner, and C. Chalopin, “Active contours driven by Cuckoo Search strategy for brain tumour images segmentation”, *Expert Systems With Applications*, Vol. 56, pp. 59–68, 2016.
7. Ye, Z, M. Wang, W. Liu, S. Chen, “Fuzzy entropy based optimal thresholding using bat algorithm”, *Applied Soft Computing*, Vol. 31, pp. 381–395, 2015.
8. Agrawal, S, R. Panda, S. Bhuyan, B.K. Panigrahi, “Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm”, *Swarm and Evolutionary Computation*, Vol. 11 pp. 16–30, 2013.
9. Horng, M and T. Jiang, “Multilevel Image Thresholding Selection based on the Firefly Algorithm”, *Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, pp. 58–63, 2010.
10. Bhandari, A. K, A. Kumar, G. K. Singh, “Tsallis entropy based multilevel thresholding for colored satellite image segmentation using evolutionary algorithms”, *Expert Systems With Applications*, Vol. 42, pp. 8707–8730, 2015.
11. Naidu, M.S.R, Rajesh Kumar. P, “Tsallis Entropy Based Image Thresholding for Image Segmentation”, *Computational Intelligence in Data Mining*, Vol. 556, pp 371-379, 2017.
12. Naidu, M.S.R, Rajesh Kumar. P, Karri. C, “Shannon and Fuzzy Entropy based Evolutionary Image Thresholding for Image Segmentation” *Alexandria Engineering Journal*, doi.org/10.1016/j.aej.2017.05.024, 2017.
13. Rényi, A, “On measures of information and entropy”, In *Proceedings of the 4th Berkeley symposium on mathematics, statistics and probability*, pp.547–561, 1960.
14. Soham S and Swagatam Das, “Multilevel Image Thresholding Based on 2D Histogram and Maximum Tsallis Entropy A Differential Evolution Approach”, *IEEE Transactions On Image Processing*, Vol. 22, No. 12, pp. 4788, 2013.
15. Passino KM (2002). Biomimicry of bacterial foraging for distributed optimization and control, *IEEE Control Systems Magazine*, 22: 52–67.
16. Kennedy J, Eberhart RC (1995). A new Optimizer using Particle Swarm Theory, in *Proceedings of Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 6: 39-43.
17. Karri. C and Jena. UR, “Hybrid gravitational search and pattern search-based image thresholding by optimizing Shannon and fuzzy entropy for image compression”, *International Journal of Image and Data Fusion*, Vol. 8, No. 3, 2017.
18. Rényi, A, “On measures of information and entropy”, In *Proceedings of the 4th Berkeley symposium on mathematics, statistics and probability*, pp.547–561, 1960.
19. Kumar N.S.R., Phanindra & Reddy P.V.G.D., Prasad. (2019). Evolutionary Image Thresholding for Image Segmentation. *International Journal of Computer Vision and Image Processing*. 9. 17-34. 10.4018/IJCVIP.2019010102.

AUTHORS PROFILE



and soft computing techniques.

N S R Phanindra Kumar obtained his B Tech (CSE) from JNTU, Kakinada in 2010 and M Tech in computer science & Engineering from AU college of engineering, Vishakhapatnam in 2013. He is working as an assistant professor in the Department of CSE, AITAM, tekkali, Srikakulam. His areas of interest include image processing



Prof. Prasad Reddy, P.V.G.D. obtained his B.Tech, in Mechanical Engineering, from Andhra University and M.Tech, in Computer Science & Technology, and Ph.D, in Computer Engineering. He is presently the Senior Professor in Computer Science & Systems

Engineering department. He has to his credit, more than 120 Research papers which includes around 95 publications in the referred indexed Journals. His Research areas include Soft Computing, Software Architectures, knowledge Discovery from Databases, Image Processing, Number theory & Cryptosystems. He has 24 years of teaching & research experience which includes more than 10 years of administrative experience with Andhra University at various senior capacities..



has 14 years of teaching and industry experience..

Dr. G. Srinivas obtained his M.Tech, in Computer Science & Technology and Ph.D in Computer Science & Engineering. He is currently working as Associate Professor in the Department of Information Technology in ANITS. His area of research is Image Processing. He