Eng. Sci., 2021, **16**, 354–365

Engineered Science

DOI: https://dx.doi.org/10.30919/es8d594

An Efficient Discrete Wavelet Transform Based Partial Hadamard Feature Extraction and Hybrid Neural Network Based Monarch Butterfly Optimization for Liver Tumor Classification

Deepak S Uplaonkar,[1,*] Virupakshappa[2] and Nagabhushan Patil[3]

Abstract

The liver tumor is one of the most widely occurring cancers nowadays. There are several forms of liver tumors, which are most often caused by hepatitis and cirrhosis. Furthermore, metastatic liver cancer may spread to other organs, posing a serious health risk. Hence it is ineluctable to diagnose this intimidating problem as early as possible. Liver tumour classification from ultrasound images is a challenging task since it is based on the structure and orientation of the liver tumour cells. To overcome this challenge, a novel hybrid artificial neural network-based monarch butterfly optimization algorithm is proposed for accurate liver tumour classification. Before the classification process, the liver tumor cells are preprocessed using different techniques such as adaptive filtering, median filtering, and color to greyscale transformation. Then the pre-processed images are segmented using the adaptively regularized kernel-based fuzzy C-means clustering algorithm and level enhanced segmentation which enhances the segmentation process and the same features are aligned in the same segment. Further, the feature vectors are extracted with the aid of the hybrid Discrete Wavelet Transform-based partial Hadamard transform method, and the same features are mapped as the same vector. The classification task is performed by a hybrid artificial neural network-based monarch butterfly optimization algorithm which enhances the classification accuracy. The comparative analyses with the state-of-art works show that the proposed work outperforms all the other approaches in terms of accuracy, specificity, sensitivity, precision, recall, and F1-score.

Keywords: Neural Network; Liver Tumor; Hybrid Discrete Wavelet Transform (DWT); Segmentation; Ultrasound Images.
Received: 03 October 2021; Accepted: 17 November 2021.
Article type: Research article.

1. Introduction

Ultrasound Image segmentation[1] became an important topic in the medical field for analyzing medical images.[2-4] For different imaging modalities, different segmentation methods are available. Meanwhile, the effectiveness of image segmentation is arduous to accomplish in ultrasound images, especially for the ultrasound liver tumor images. Significantly, the ultrasound image segmentation purely depends on the quality of data. Few of the factors that have impacts on the ultrasound image segmentation are attenuation, speckle and signal dropout.[5]

Liver tumor[6] is the most commonly affected tumor among men which leads to death. It has been caused due to viral hepatitis. Hence it is necessary to segment the liver tumor for medical applications such as computer-aided surgery, diagnosis of hepatic diseases, estimation of liver volume, and planning for treatment.[7] Most probably the medicos utilize computed tomography (CT) scans for the liver tumor segmentation, 3D rendering, and consequently liver vasculature analysis. There are several studies[8-10] that have been carried out though still, it is a crucial one since the non-rigid structure of the liver tumor and ill-defined borders feature the CT images.

For further enhancement, the medical imaging sector utilizes contrast-enhanced computed tomography (CECT) for liver imaging and monitoring.[11] The segmented images are

---

[1] *PDA College of Engineering, Kalaburagi 585101, Karnataka, India.*

[2] *Department of Computer Science and Engineering, Sharnbasva University, Kalaburagi 585101, Karnataka, India.*

[3] *Department of Electrical and Electronics Engineering, PDA College of Engineering, Kalaburagi 585101, Karnataka, India.*

**Email*: uplaonkar@gmail.com (Deepak Uplaonkar)

used to identify the exact size and positions of the tumors inside the liver cells. The classification of normal and tumor cells from ultrasound liver images has been the subject of extensive research.[12] However, due to the irregular structure and low contrast borders of the tumor cells, it became a difficult process. To tackle this entire problem, we proposed a novel method known as a hybrid artificial neural network-based monarch butterfly optimization algorithm.[13] Exploiting this approach, we obtain maximum accuracy of classification. Our work's contributions are listed below,

- Then the pre-processed images are subjected to segmentation by utilizing the adaptively regularized kernel-based fuzzy C-means clustering algorithm (ARKFCM) and level enhanced segmentation.
- Henceforth the feature vectors from the segmented images are derived by hybrid DWT-based Partial Hadamard Transform for the further classification process.
- The classification process is performed with the exploitation of a hybrid neural network-based monarchy butterfly optimization algorithm which obtained optimized classification outcomes.

The remaining article is arranged as follows. Section 2 explains the related works. Section 3 elucidates the proposed methodology in a wider context. The experimental analyses are illustrated in section 4. Finally, we concluded our work in section 5.

2. Related Work

Devi *et al.*[14] developed an SVM-based classifier and feature difference technique for the automatic detection of liver tumors. For semantic pixel-wise categorization, a deep learning-based technique has been developed. The novel contrast-oriented feature-difference technique performs the malignant and benign liver lesions classification. The higher detection accuracy is obtained but the computational complexity is higher.

Pan *et al.*[15] introduced an attention mechanism with a bidirectional Long Short-Term Memory (LSTM) neural network (A-bLSTM) for liver cancer detection. To gain more detail from ultrasound images, dense connectivity integrates low-level fine segmentation details with high-level coarse segmentation details. This research incorporates factorized filters into the network to improve the model's performance. This method is time-consuming and most prohibitive.

Sureshkumar *et al*.[16] introduced deep learning (DL) technique to enhance liver tumor classification accuracy. The segmentation of liver tumors is a necessary step before any surgical intervention. Compared with different methodologies, this deep learning model outperformed better classification accuracies, nevertheless, the computational complexity and time execution are higher.

The harmony search algorithm with hybrid deep learning architecture (HSA-HDL) was introduced by Baneamoon *et al*.[17] for liver tumor detection. The evaluation and accuracy of liver tumor diagnosis are improved for a given image. The experimental results are evaluated using a large number of liver images obtained from National Center for Toxicological Research liver cancer database (NCTRlcdb). Hence, the experimental results demonstrated higher effective identification thereby delivering effective analysis but the accuracy result are lower.

Based on the clinical data and magnetic resonance imaging, Zhen *et al.*[18] proposed deep learning model for accurate detection of liver cancer. The three-way malignancy-classifiers, binary classifiers and seven-way classifiers are trained using Convolutional Neural Network (CNN) in which the data obtained from 1,210 patients with liver tumors. The three experienced radiologist's performances with the sensitivity and specificity of models performance is improved. The economical cost is reduced but takes larger execution time.

The genomic alterations and correlation with immune suppression model was proposed by Fujita *et al.*[19] for primary liver cancer classification. From a Japanese population, immune gene signatures of 234 main liver cancers are characterized. The regulatory T cell (Treg), cytolytic activity (CYT), Catenin Beta 1 (CTNNB1) and tumor associated macrophage (TAM) class of liver cancer are classified. The chromatin regulator AT-rich interaction domain 2 (ARID2) of somatic mutations is associated to the eminent stages of extracellular matrix genes. This method failed to provide better accurate results.

The Deep Convolutional Neural Network (DCNN) was suggested by Zeng *et al*.[20] for Liver tumor classification. The segmentation structure of the tumor is obtained and the learned the features in which the training data extract the various sizes of image data blocks. The region mutual exclusion constraints with the model of multi-region segmentation are established. The regional adhesion and boundary blur finding is complex it may cause the boundary point attribution area. This method demonstrated higher accuracy with higher processing time.

From the literature analysis, we have find out few issues such as higher computational complexity, higher time-consumption, most prohibitive, lower accuracy and higher processing time. To tackle these issues, a novel hybrid artificial neural network-based monarch butterfly optimization algorithm is proposed for accurate liver tumour classification.

3. Methodology

The liver tumor is the most commonly occurring tumor in both men and women nowadays. To analyze the depth of the tumor, the ultrasound images are classified by the medicos. Prior to this, the images are segmented by the radiologists manually via a slice-by-slice analysis on various ultrasound images. Significantly, this manual approach consumes a significant amount of time and is complex in nature. Meanwhile, the segmented images are fed as an input to the computerized diagnosis and hence it minimizes the efficacy dramatically.

However, the automatic segmentation of liver tumorimages is also crucial due to the facts that the (i) liver is attached to the other surrounding organs where, those organs to exhibits the same ultrasound value as the liver, (ii) it is arduous to differentiate the liver tumor region and normal region due to the brightness of the ultrasound images and also blurred boundaries of liver tumor, and (iii) the shape, size, and position of liver tumors are wavering and hence the automatic segmentation by the machine are critical. Fig. 1 depicts the proposed liver tumor segmentation and classification process. From Fig. 1, it is observed that the proposed methodology involves four stages: i) image pre-processing, (ii) Liver tumor image segmentation, (iii) Feature extraction, and (iv) classification. The following sections reveal the four stages in a wider context.

3.1 Image Preprocessing

In the first work, we have used Contrast Limiting Adaptive Histogram Equalization (CLAHE) algorithm for image pre-processing and image enhancement. Both colored and gray scale images noises removed using a "clip-limit" function. In the second work, the CLAHE methodology is applied to pre-processes the image after the acquisition of liver ultrasound images.

In this study, we have the pre-processing step is carried out by performing color to grayscale transformation, adaptive filtering and median filtering techniques. To begin with, the collected ultrasound images are pre-processed to transform the source images into the required image format. The main aim of the pre-processing step is to enhance the liver tumor image features by suppressing the unwanted distractions and hones up the important features for further processing. In our proposed method we have adopted three techniques for suppressing unwanted features and at the same time improves the features that to be extracted.

(i) Color to grayscale transformation,
(ii) Adaptive filtering, and
(iii) Median filtering.

3.1.1 Color to grayscale transformation

The color to grayscale transformation is performed to enhance the image quality by improving the speed, and scalability.

3.1.2 Adaptive filtering

During image pre-processing, the adaptive filters[21] are used as a denoiser which removes the noise without affecting the quality of the image. Adaptive filters are mostly managed by the parameters. Hence, the orientation controlled adaptive filter can be expressed as,

$$AF_{adap} = AF(\phi, Q) \tag{1}$$

Here, ϕ represents the orientation parameter and Q is the anisotropy.[22] In the source image, the noise can be removed by deeming orientation evaluation as the starting parameter.

3.1.3 Median filtering

After employing the de-noising method, it is ineluctable to maintain the spatial characteristics of an image. Hence, we have adopted a non-linear smoothing filter known as median filter[23] to preserve the edges in the image from blurring. The exploitation of median filter can help to replace the required point by the brightness that was attained from its adjacent edge. This because the noises do not affect the median of the brightness of the adjacent edge. Meanwhile, the median filter also removes the impulse noise. Thus the output of the median filter is the enhanced quality image of the ultrasound liver tumor image. The resultant output is expressed as,

$$I_M(a,t) = med\{I(a-u, t-v)u, v \in M\} \tag{2}$$

Here, I_M & I denotes the median filtered and original images respectively. H is the 2D mask used for median filtering. However, the final output is subjected to segmentation for further process.

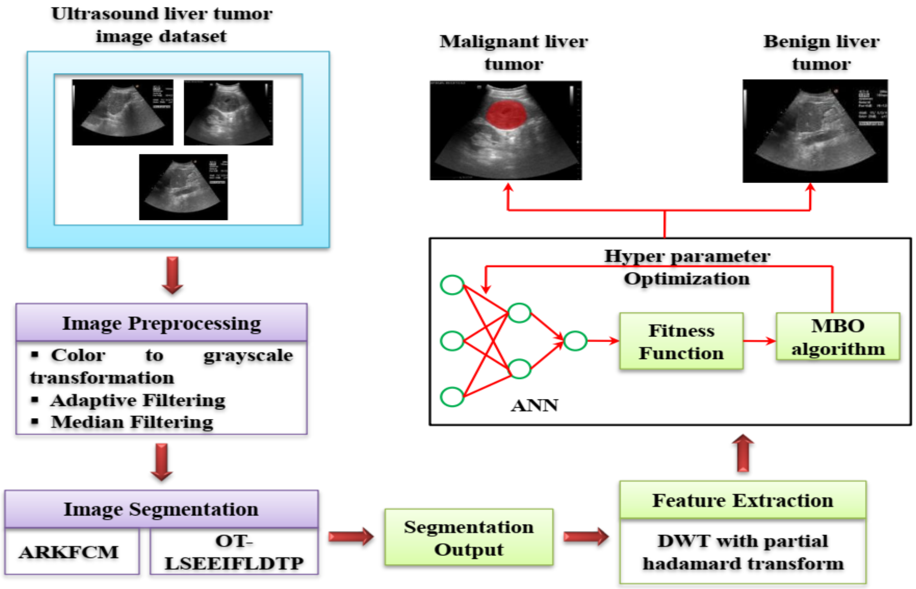

Fig. 1 Proposed architecture.

3.2 Liver tumor image segmentation

In the first work, the adaptively regularized kernel-based fuzzy C means (ARKFCM) tumor segmentation. The selective level set approach and local ternary pattern with ARKFCM segment the image. The liver tumor image segmentation can be conducted ARKFCM algorithm[24] and level enhanced segmentation. The Fuzzy C Means (FCM) algorithm allocates a membership value to every pixel present in the cluster in the image space. In the second work, the local directional ternary pattern technique with enhanced edge indicator function and Otsu thresholding-based level set model is proposed.

In this study, we proposed an adaptively regularized kernel-based fuzzy C-means clustering algorithm and level enhanced segmentation for segmentation process. For an image *X*the image space is formed with a set of grayscales g_t at pixel t (t=1,2,…,N), A={a_1,a_2,…,a_N}$\in R^l$ in l dimensional space, cluster centers $c = \{c_1, c_2, \dots, c_p\}$, where p is a positive integer $(2 < p << N)$. Each pixel t in the k^{th} cluster has a membership value m_{tk}. The FCM algorithm's objective function is calculated as follows:

$$J_F = \sum_{t=1}^{N} \sum_{k=1}^{p} m_{tk}^{w} \|a_t - c_k\|^2, \tag{3}$$

where w is the fuzziness weighting exponent, and w>1. The $\|a_t - c_k\|^2$is the grayscale Euclidean distance.

The membership value $m_{tk}$is limited using the following conditions:

$$\forall t \in [1, N], k \in [1, p] \tag{4}$$

$$\sum_{k=1}^{p} m_{tk} = 1, m_{tk} \in [0,1], 0 \le \sum_{t=1}^{N} m_{tk} < N \tag{5}$$

Utilizing the alternate optimization, the cluster center (c_k) and membership function (m_{tk}) are updated as follows:

$$m_{tk} = \frac{1}{\sum_{l=1}^{p} (\|a_t - c_k\|^2 / \|a_t - c_l\|^2)^{1/(w-1)}} \tag{6}$$

$$c_k = \frac{\sum_{t=1}^{N} m_{tk}^{w} a_t}{\sum_{t=1}^{N} m_{tk}^{w}} \tag{7}$$

The FCM technique is also susceptible to noise, which decreases the accuracy of the clustering technique and reduces image artifacts and it is mainly because the objective function in equation (3) does not use any local information. To overcome this problem, the regularization parameter is included in this work to improve the segmentation and keep the image details intact. The regularization parameter used to enhance the accuracy is Gaussian radial basis function (GRBF). To control the amount of contextual information used, the regularization term τ is used. The noise level varies from one window to another, hence we are incapable of using a fixed value for τ for every pixel. Based on the pixel being processed, the τ value is adaptively computed.

The Local Variation Coefficient (LVC)[25] is first measured and normalized concerning the local average grayscale to determine the discrepancy between grayscales in the local window. The value of LVC increases in the presence of noise, resulting in greater variability between the central pixel and its neighbors. Assume

$$L_{var\_coef_t} = \frac{\sum_{l \in N_t} (a_l - \bar{a}_t)^2}{N_C * (\bar{a}_t)^2}, \tag{8}$$

where a_l is the grayscale value of the pixel in the local window, L_{var_coef} represents LVC, N_C is the cardinality value of N_t, and $\bar{a}_t$is the mean grayscale value. To infer the weights within the local window, $L_{var_coef_t}$is applied to the exponential function as shown below:

$$\rho_t = exp\left(\sum_{l \in N_t, t \neq l} L_{var\_coef_l}\right) \tag{9}$$

$$v_t = \frac{\rho_t}{\sum_{l \in N_t} \rho_t} \tag{10}$$

$$\phi_t = \begin{cases} 2 + v_t, & \bar{a}_t < a_t \\ 2 - v_t, & \bar{a}_t > a_t \\ 0, & \bar{a}_t = a_t \end{cases} \tag{11}$$

The parameter ϕ_t assigns greater values to pixels with large LVC values and allocates smaller values for pixels with lower LVC values. If the values of the central pixel grayscale and local average grayscale are the same, then the algorithm behaves like a conventional FCM algorithm making the value of ϕ_t as 0. To preserve the image details and balance the convergence rate the value of ϕ_t to 2.

The weighted image is generated by forming $\bar{a}$ which is the grayscale value of the average actual image filter and it can be replaced with the recently created weighted image $\bar{v}_t$ and it is derived using the below equation (12):

$$\bar{v}_t = \frac{1}{2+max(\phi_t)}\left(a_t + \frac{1+max(\phi_t)}{N_C - 1} \sum_{j \in N_t} a_j\right) \tag{12}$$

where a_j and N_t represents the grayscale and adjacent pixels *i*. The above formula is mainly used to create the weighted image using ϕ_t to make the image that is free of parameters that are difficult to change. To balance the contextual details, the ARKFCM structure is developed by comparing the adaptive regularization parameter $\phi_t$associated with each pixel. Equation (13) illustrates how the ARKFCM's objective function is calculated:

$$J_A = 2\left[\sum_{t=1}^{N} \sum_{k=1}^{p} m_{tk}^{w} \left(1 - K(a_t, c_k)\right) + \sum_{t=1}^{N} \sum_{k=1}^{p} \phi_t m_{tk}^{w} \left(1 - K(\bar{a}_t, c_k)\right)\right] \tag{13}$$

$$m_{tk} = \frac{\left(\left(1 - K(a_t, c_k)\right) + \phi_t\left(1 - K(\bar{a}_t, c_k)\right)\right)^{-1/(w-1)}}{\sum_{l=1}^{p} \left(1 - K(a_t, c_l)\right) + \phi_t\left(1 - K(\bar{a}_t, c_l)\right)} \tag{14}$$

$$c_t = \frac{\sum_{t=1}^{M} m_{tk}^{w} (K(a_t, c_k) + a_t) + \phi_t (K(\bar{a}_t, c_k) \bar{a}_t)^{-1/(w-1)}}{\sum_{l=1}^{p} m_{tk}^{w} \left(K(a_t, c_k)\right) + \phi_t \left(K(\bar{a}_t, c_k)\right)^{-1/(w-1)}} \tag{15}$$

When the value of $\bar{a}$ is replaced with the grayscale value of the original image's average/median filter, the ARKFCM algorithm is defined as ARKFCM-I/ARKFCM-II. The algorithm is defined as ARKFCM_v if is replaced with the weighted image provided by equation (12). The ARKFCM algorithm is formulated using the following steps:

Step-1: Initialize threshold value T=0.01, w=2, loop count l_c=0,c, and $m^{(0)}$.

Step-2: Compute the adaptive regularization parameter ϕ_t.

Step-3: Compute the value $\bar{a}$ to generate algorithm ARKFCM-I/ ARKFCM-II and compute the value $\bar{v}$ to generate algorithm ARKFCM_v

Step-4: Compute the cluster centers $c_t^{(l_c)}$ and $m^{(l_c)}$ using equations (14) and (15)

Step-5: If maximum $\|m^{(l_c+1)} - m^{(l_c)}\| < Torl_c > 100$ then terminates the process or else update the value of $l_c$as $l_c = l_c + 1$ and execute step-3.

3.2.1 Selective level segmentation with ARKFCM algorithm

The selective level set segmentation[26,27] starts with an initial interface ψ0 and ends with a locally optimal site ψe used for segmenting the liver tumor images. The ARKFCM algorithm deals with inaccuracy and inconsistencies present in the image. It divides the image constituents into N classes, with equal components in one class and dissimilar components in the other. In the Hamiltonian-Jacobi level set model(HJ-LSM),[28] a constant balloon force(b0) is associated with the image gradient that is either drawn or directed towards the dynamic interface. The object indication function is used to modulate the constant force for image segmentation where the value near boundaries is zero and the value near the homogeneous regions is higher. The convention model proposed by Chan and Vese[29] is formulated as shown below:

$$K = \int_{\sigma} i \bullet H(\psi) dadb - \int_{\sigma} i \bullet (1 - H(\psi)) dadb \tag{16}$$

In the above equation (16), i represent the intensity variance and $H(\psi)$ is the Heaviside function. The value of K is often high due to the presence of noise and inhomogeneity which will often lead to level set evolution. For selective region competition, a new term is presented as follows:

$$K = \sum_{r \in R} \alpha_r - \sum_{(i \in M) \cap (i \notin R)} \alpha_i \tag{17}$$

$$\psi_0 = 2\left(\sum \alpha_r > c\right) - 1 \tag{18}$$

The parameter α_r is similar to the value derived from the above equation which represents the components derived using the ARKFCM algorithm. The components left out from the ARKFCM algorithm are represented using the parameter α_i. The parameter c is a customizable parameter whose value falls between 0 and 1 and it is set to 0.5 in this work for efficiency. The fuzzy region force value K varies between -1 and 1. The sign mainly represents the augmentation and contraction of the dynamic interface. To move the dynamic interface towards the area of interest, the benefits of the ARKFCM clustering along with a signed balloon force is formed as follows:

$$O = \left[1 - \beta\left(2 \sum \alpha_r - 1\right)\right] b_0 \tag{19}$$

The tradeoff parameter $\beta (0 \leq \beta \leq 1)$ utilizes two conditions: If β =0, the value of the balloon force(b_0) is constant. If β =1, the value of the balloon force(b_0) is altered using the selective ARKFCM membership function $\sum \alpha_r$. The balloon force output O is a matrix that comprises a parameter that moves forward or drags the force towards each pixel.

To overcome the boundary leakage problem, a standard object indication function is presented to enhance a faster convergence as shown below:

$$W = e^{-10\, max(\gamma . n_j, (1-\gamma) n_\alpha)} \tag{20}$$

The parameter γ balances the tradeoff between different object indication functions and to improve the convergence of the level set evolution a constant value of 10 is used. n_α is a term that mainly occurs as a result of the membership function present in the ARKFCM algorithm. The normalized edge indicator n_j for the image gradient is presented as follows:

$$n_j = \frac{n - min(n)}{max(n)} \tag{21}$$

where the value n indicates

$$n = \frac{1}{1 + |\nabla (G_k * Z)|^2} \tag{22}$$

The above equation (22) mainly generates a convolution between the Gaussian kernel G_k and the image Z.

The above solutions can be merged to create a selective level set segmentation process as follows:

$$\begin{cases} \frac{\partial \psi}{\partial e} = \delta(\psi)[\lambda W \bullet O + (1 - \lambda) K] \\ \psi(a, b, e = 0) = \psi_0(a, b) \end{cases} \tag{23}$$

In the above equation (23), δ is the Dirac function of the dynamic interface, and the coordinating parameter is represented as λ. The interface should be close to the signed distance function when level set evolution is taking place. Hence, the dynamic interfaces are re-initialized in periodic sequences for the signed distance function. The interface evolution is conducted as shown below:

$$\begin{cases} \frac{\partial \psi}{\partial e} = \delta(\psi)[\lambda W \bullet O + (1 - \lambda) K] \\ \psi = G_k * \psi \end{cases} \tag{24}$$

The convolution operation is represented using a* and the Gaussian smoothing kernel is represented as G_k.

The selective area (tumor regions) in the liver ultrasound images is segmented using the level set steps provided below:

Step 1: Execute the steps in the ARKFCM algorithm

Step 2: Select the important region of interest[α_r]

Step 3: Calculate the improved object indication function W from equation (20) and signed balloon force O from equation (19).

Step 4: Initialize the value for the dynamic interface using equation (18) and assign it to ψ.

Step 5: Calculate H(ψ), δ(ψ), Δ(ψ), and $\nabla(\psi)$.

Step 6: Compute the ARKFCM algorithms region force value *K*.

Step 7: Improve and standardize the dynamic interface value as shown in equation (24)

Step 8: If the end condition is not met, Go to steps 5 and repeat the process.

3.3 Feature extraction

The steps involved in the derivation of features from each segmented image are disclosed in this section. Generally, the DWT performance with few inacurate results during feature extraction. Hence, we added DWT with partial Hadamard transform for better results in terms of feature extraction. The features are derived by the utilization of hybrid DWT-based Partial Hadamard Transform. These derived features can be applicable for translation, rotation, and scaling. Besides two-level DWT[30] is used to process the segments and following this employed partial Hadamard transform[31,32] to derive the multidirectional attributes from the low-frequency subband. At the same time, DWT can be used to derive the features of a segmented image by sequential low and high pass filtering. Thus, the DWT can be expressed as,

$$h(x) = \sum_k a_{i_0}(k) \varphi_{i_0,k}(x) + \sum_{i=i_0}^{\infty} \sum_k d_i(k) \psi_{i,k}(x) \tag{25}$$

The wavelet coefficients and the coarser signal approximate expanded coefficient can be indicated as $d_i$anda_{i_0}. Meanwhile, the basic function that is derived from the translation and dyadic dilation process is represented as, $\varphi_{i_0,k}(x)$ and $\psi_{i,k}(x)$ can be expressed as,

$$\varphi_{i_0,k}(x) = 2^{i/2} f_i(2^i x - k) \quad (26)$$

$$\psi_{i_0,k}(x) = 2^{i/2} g_i(2^i x - k) \quad (27)$$

Consequently, the translation and dilation parameters are represented as k and high pass and low pass filter coefficients for the liver tumor image I_t is determined as f_i and g_i respectively. After the employment of wavelet coefficients, the partial Hadamard transform is applied. The partial Hadamard transform is explained using the following processes. A row vector of length m is represented as $A = [a_1, a_2, \ldots, a_m]$.

Here $m<Z$ and $Z=2^z$ (Z is the length of the vector x_D) and all the variables present in A are strictly positive integers with $a_i \neq a_j$ for every $i \neq j$. The vector A can be randomly generated and plays a key role in the feature extraction process to leave out the unimportant features. Using A, a submatrix for the Hadamard matrix (H_z) is represented as P_z and the i^{th} row of P_z is the a_i^{th} row of H_Z for $i\in [1,2,\ldots.,m]$. Now the size of the P_Z is computed as $m \times z$. The partial Hadamard transform shown below can be performed when once P is formed using the random vector A.

$$x_H = P_z x_D \quad (28)$$

The complex value vector x_H is formed by satisfying the Hadamard transform properties. The Hadamard transform needs to satisfy the following properties:

- Every entry equal± 1.
- Any two rows of the Hz matrix are orthogonal
- Hz is both symmetric and orthogonal.

The transformations in a Hadamard matrix are non-invertible and P_Z is a row reduced submatrix of H_Z which is always a column rank deficient. For equation (28), there exist multiple solutions based on the linear algebra point of view. The values of the features extracted using the DWT-Partial Hadamard transform fall in the range +1 or -1. Since equation (28) includes no multiplication operation, the partial Hadamard transform involves less computational complexity and can be implemented using different Hadamard transform structures. Using the hybrid-DWT-based partial Hadamard transform, the feature vectors of each segment of the tumor are computed. After the features for each segment is extracted it should be matched to one another to identify the matched segments.

3.4 Classification

The classification process is carried out with the aid of a hybrid artificial neural network-based Monarchy Butterfly Optimization algorithm.

3.4.1 Artificial neural network

An artificial neural network[33,34] is made up of input, hidden, and output layers. Any of these may have more than one hidden layer. All the layers are linked by using a connection known as weight (numeric number). The output layer can be expressed as,

$$N_O = \chi\left(\sum_{j=1}^{N} WN_{ij}x_j + TN_i^{hid}\right) \quad (29)$$

Here, function $\chi()$is used to denote the activation function or transfer function. N is the total number of neurons and WN_{ij} is used to represent the weights and x_j is the input neuron. $TN_i^{hid}$is the threshold terms that are used to note the hidden neurons. The activation function is used to introduce the nonlinearity function. An example of activation function is sigmoid function and can be expressed as,

$$\chi(u) = \frac{1}{1+exp(-u)} \quad (30)$$

There are different types of activation functions but the simplest and easiest one is the sigmoid function. Most probably, the Artificial Neural Network (ANN) is the numerical indication of human neural architecture which reflects human knowledge and aptitudes. More often the automatic liver tumor diagnosis is an intricate one and supports manual diagnosis which is a time-consuming process and hence ANN is the better option for the replacement of manual operations. This approach classifies the liver tumor cell accurately within a short period of time and thus minimizes the time consumption. For this reason, we adopted an artificial neural network for the classification process.

Further to enhance the classification accuracy we have taken an optimization algorithm known as the Monarchy Butterfly optimization (MBO) algorithm. The ANN is trained using the MBO algorithm which minimizes the error rate by adjusting the weights and biases of the ANN appropriately.

3.4.2 Monarch butterfly optimization algorithm

Monarch butterfly optimization (MBO's) approach initiates the migration activity of monarch butterflies.[35,36] Every monarch butterfly is found in two countries, one in the north of the United States and one in the south of Canada. The offsprings are generated using the migration operator in the initial stage. The migration ratio adjusts the total number of offsprings. Hence, the other butterflies' positions are determined via the butterfly adjusting operator. Update the monarch butterflies position using the migration operator that generates the offsprings. It is adjusted using the ratio of migration. Tune the positions for other butterflies by means of butterfly adjusting operator. The fitness evaluations are minimized and in order to keep the population unchanged. The land one (L_1) and land two (L_2) are the two subpopulations of individual monarch butterflies, which is denoted as below:

$$L_1 = P_1 \times Ceil(r \times P) \quad (31)$$

$$L2 = P \times P \times (P_1) \quad (32)$$

From the above equations (31) and (32), the operator rounds are *ceil(a)* in which '*a*' value greater than or equal to'*a*'. The migration ratio of L_1 is r and the total amount of monarch butterfly population is P. The new child population is generated L_1 and L_2 in monarch butterflies.

The population is kept as constant in which the value of parent butterflies is forever better than the child. If the child remains unchanged, the population will continue to grow, and successful individuals will emerge. The below equation (33) formulate this concept.

$$B_{t,k}^{g+1} = B_{b_1,k}^{g} \tag{33}$$

At the g+1thgeneration, the kth position (B) of the butterfly t is represented as $A_{t,k}^{g+1}$. For each monarch butterfly b1, the kth updated components are denoted as $A_{b_1,k}^{g}$. Where g is the number of iterations. When the value of L1 is less than or equal to r, the value of b is modified using the equation (34) below.

$$b = \gamma \times \lambda \tag{34}$$

where the time in use for migration is$\lambda$and the uniform and random manner is expressed as γ. If the value of r is less than b, the element k for the new monarch butterfly is expressed using the equation (35) below.

$$B_{b,k}^{g+1} = B_{b_2,k}^{g} \tag{35}$$

For each butterfly b2, the kth updated element B_{b_2} is denoted as$B_{b,k}^{g+1}$. While the value of r is bigger, there is a logical balance between L1 and L2.[37] If p is big, more elements from monarch butterflies in L1 will be selected. If p is small, more elements from monarch butterflies in L2 will be selected. If the child value for butterfly t is less than or equal to r, the butterfly location is modified using the following equation.

$$B_{t,k}^{g+1} = B_{best,k}^{g} \tag{36}$$

where, the k^{th} individual of B_{best} is denoted as $B_{best,k}^{g}$ thereby providing optimal population result. The position is updated when the r-value is lesser than λ.

$$B_{t,k}^{g+1} = B_{b_3,k}^{g} \tag{37}$$

$$b_3 \in [1,2,\ldots\ldots\ldots, P_2] \tag{38}$$

From L_2, the k^{th} arbitrarily chosen monarch butterfly B_{b3} is$B_{b_3,k}^{g}$. Update the position value if the butterfly adjustment rate (B_{adj}) is less than the valueγ.

$$B_{t,k}^{g+1} = B_{t_3,k}^{g} + \beta \times (dB_k - 0.5) \tag{39}$$

In the above equation (39), the monarch butterfly walking step is represented as *dB*. The value of *dB* is computed as shown below:

$$dB = Levy(B_t^g) \tag{40}$$

The weighting coefficient is indicated using β value as shown in equation (41).

$$\beta = \frac{mw}{g^2} \tag{41}$$

The maximum walking step taken by the butterfly is expressed as *mw* for a single step. Create the long search step that increases toβ. If the value of β is high then the algorithm enters into the exploration phase which improves the influence of *d B*in $B_{t,k}^{g+1}$.

If the value of β is low, then the algorithm enters into the exploitation phase. The exploitation stage is resultant while allocating the value β is small. Algorithm 1 demonstrates the MBO algorithm.

3.4.3 Formulation of hybrid artificial neural network-based monarchy butterfly optimization algorithm

The liver tumor classification framework is presented in Fig. 2. In the initialization stage, the input and output vector of the liver tumor classification problem is evaluated. Based on the problem, the ANN structure is built. The structure mainly comprises the number of neurons in each layer and the number of layers in the ANN. The weights of the links that connect two neurons, and the output threshold values of the hidden nodes are also considered crucial features. The position vector of the butterfly in the Monarchy Butterfly Optimization (MBO) algorithm is one of the important parameters of the ANN network. Each butterfly in the population represents the neural network's initial configuration. Even though the ANN is capable of modeling complex data in a non-linear manner, its output is largely determined by the initial parameter setting.[38] The MBO algorithm is used to set the ANN network's initial parameters. The ANN is trained to produce a final prediction model using the backpropagation algorithm. Each butterfly in the subpopulation is converted into an ANN parameter configuration during the optimization process, and the ANN is then trained using the training data. The butterfly with the lowest prediction error is selected as the global best butterfly. After the end condition is met, the global best solution obtained is used as the ANN network's initial configuration.[39] The number of input and output neurons corresponds to the amount of input and output data. The ANN formed using the MBOs optimal topology in this work comprises four inputs, one hidden layer with five neurons, and one output layer with two neurons(4-5-2). The population size of the MBO algorithm is set to 10 and the algorithm is run a maximum of 10 times to achieve the enhanced accuracy with less error rate.

Algorithm 1: Monarch Butterfly Optimization (MBO) algorithm

```
Start
Initialize the MBO parameters with the maximum number of iterations
While (g<Max_g)
    Evaluate the fitness function
    Generate the subpopulations as L1 and L2
    Equation (1) is used to generate subpopulation L1
    Equation (2) is used to generate subpopulation L2
    Combine the novel created sub-populations
    The current optimal individual is determined
End while
Determine the global most excellent individuals
End
```

4. Result and discussion

The proposed HNNMBO method performance is evaluated in this section. The proposed methodology is implemented using MATLAB 2018a software. To evaluate the efficiency of the proposed work for liver tumor classification, various varieties of evaluation criteria with state-of-the-art comparison are used.

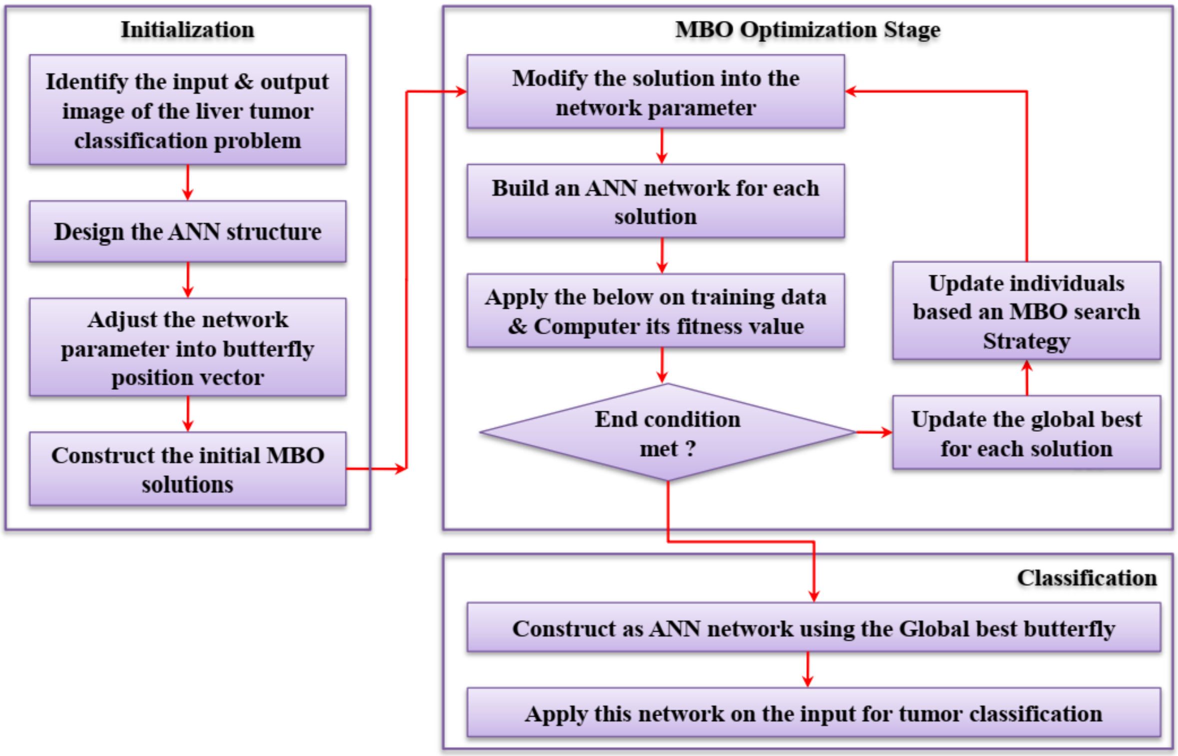

Fig. 2 Hybrid ANN-MBO approach for liver tumor classification.

4.1 Dataset explanation

The experimental images were obtained from the Liver Tumor Segmentation (LiTS) dataset.[40] There are three different types of tumors such as metastatic carcinoma, hepatocellular carcinoma and meningioma with patients details were included in this dataset. For testing the model, consider 70% of the data and rest of 30% for testing process. Five-fold cross-validation is used for the training and testing process.[41] Under image enhancement, resize the images into 180 × 180 resolution. Without losing important information, the dimension reduction is performed to reduce the computation and training time. The comparative analysis is conducted by executing every method on the LiTS dataset using the Matlab programming language.

4.2 Performance measures:

The liver tumor classification using the proposed method is evaluated via various measures including accuracy (A), specificity (Spe), sensitivity (Sen), precision (P), false positive rate (F_{PR}), false negative rate (F_{NR}), false discovery ratio (F_{DR}), negative predictive value (N_{PV}) and F-score.

4.3 Performance analysis

Table 1 demonstrates the segmentation result. This experimental investigation is conducted between different methods for segmentation result analysis. Two segmentation techniques namely ARKFCM with an enhanced level set[23] and Otsu thresholding-based level set with enhanced edge indicator function and local directional ternary pattern technique (OT-LSEEIF-LDTP)[24,25] are applied to the proposed technique to test its efficiency. In OT-LSEEIF-LDTP, initially, the contrast limited adaptive histogram equalization technique is applied which increases initial image contrast while maintaining image brightness and also improves the accuracy of liver Ultrasound images. Following image enhancement, a local directional ternary pattern technique and an Otsu thresholding-based level set with an enhanced edge indicator function are utilized for segmenting the liver lesion/tumor region.

Table 1. Segmentation result.

Methodology	Dice coefficient
Graph-based algorithm	94%
Empirical greedy machine	87%
ACS-AOT	92%
ARKFCM with enhanced level set	97.11%
OT-LSEEIF-LDTP	99.43%
ARKFCM with enhanced level set with the hybrid neural network with MBO	99.32%
OT-LSEEIF-LDTP with the hybrid neural network with MBO	99.87%

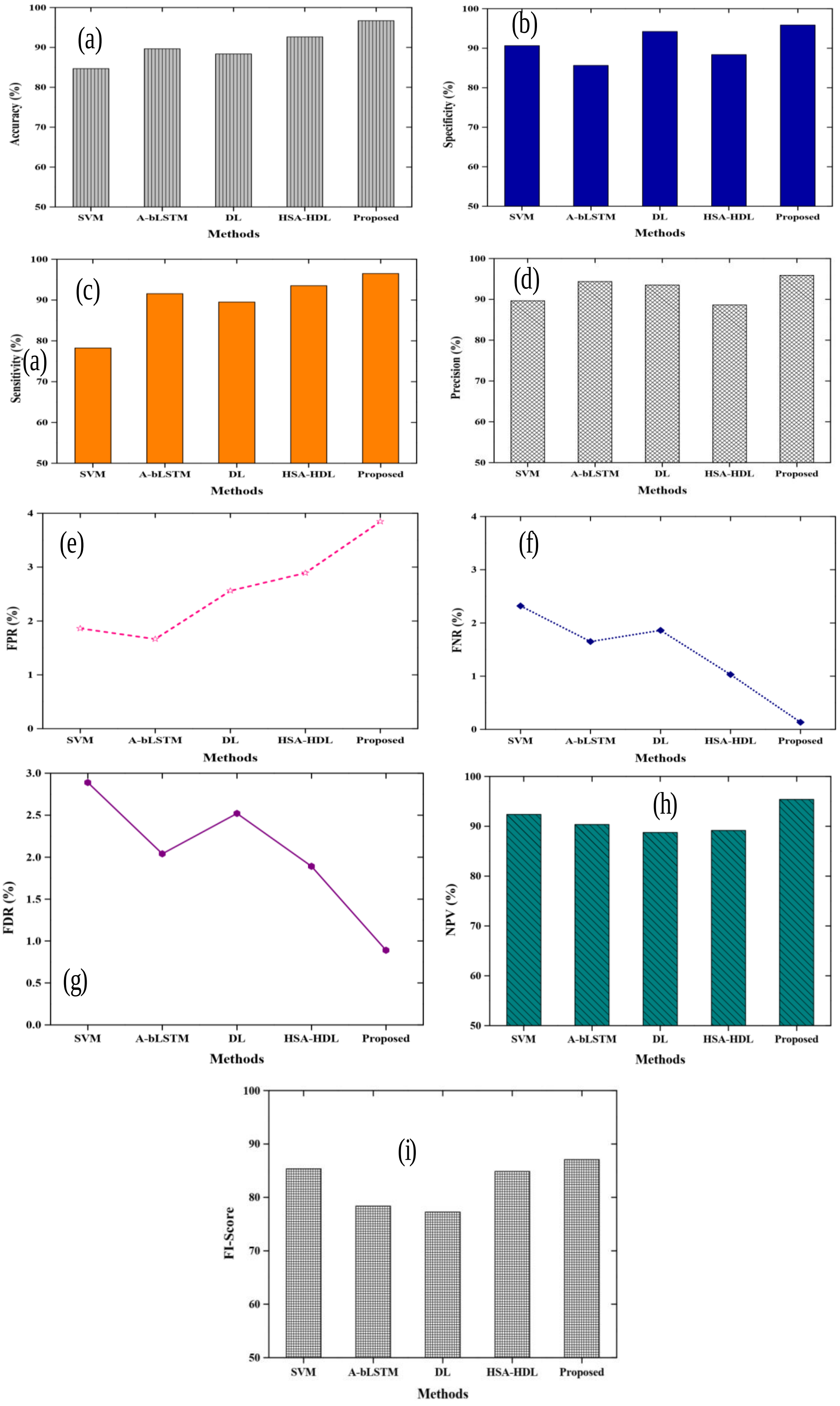

Fig. 3 State-of-art performance analysis (a) accuracy, (b) specificity, (c) sensitivity, (d) precision,(e) false positive rate, (f) false negative rate, (g) false discovery ratio, (h) negative predictive value and (i) F-score.

From this investigation, the graph-based algorithm, empirical greedy machine, active contour segmentation with adaptive Otsu based thresholding (ACS-AOT), ARKFCM with an enhanced level set, OT-LSEEIF-LDTP, ARKFCM with enhanced level set with the hybrid neural network with MBO. and OT-LSEEIF-LDTP with the hybrid neural network with MBO provided 94%, 97.11%, 99.43%, 99.32% and 99.87% segmentation results.

Fig. 3 explains the state-of-art performance analysis with respect to different evaluation measures such as accuracy, specificity, sensitivity, precision, false-positive rate, false-negative rate, false discovery ratio, negative predictive value, and F-score. Figs. 3(a) to 3(i) represents the classification evaluation of the proposed model with various conventional approaches including Support Vector Machine (SVM), A-bLSTM, Deep Learning (DL), Harmony Search Algorithm based Hybrid Deep Learning (HSA - HDL) and the proposed method. From Fig. 3(a), the proposed accuracy percentage is increased to 96.69%. The SVM, A-bLSTM, DL, HSA-HDL and proposed method accomplished 83%, 90%, 89%, 92% and 96.69% respectively. Nevertheless, the proposed method demonstrated higher accuracy than other existing methods. The specificity and sensitivity results are 95.85% and 95.89% by using the proposed method from Figs. 3(b) and 3(c).

According to Figs. 3(d) and 3(e), the precision and false positive rate is 95.89% and 3.846%. Further, the proposed method yielded a 0.134% false-negative rate and 0.89% false discovery ratio according to the investigation as shown in Figs. 3(f) and 3(g). The proposed technique demonstrates 95.38% negative predictive value and 87.09% F-score values than other methods. Anyway, the proposed method accomplished superior performances than other existing methods.

Fig. 4 explains the computational complexity. This experiment is conducted between different existing methods such as SVM, A-bLSTM, DL, HSA-HDL and proposed with respect to computational time as mentioned in seconds. The computational complexity of SVM, A-bLSTM, DL, HSA-HDL and proposed method is 1.244s, 1.575s, 1.8936s, 3.2945s and 1.0099s. Anyway, the proposed method takes minimum computational time than another method.

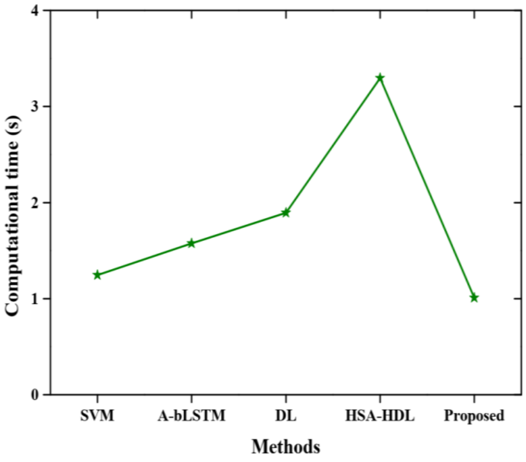

Fig. 4 Computational complexity analysis.

Table 2 explains the five-fold cross validation results. Further, the one-to-five-fold validations with respect to Matthews correlation coefficient (MCC), accuracy, specificity and sensitivity results are obtained as delineated in Table 2.

Table 2. Five-fold cross validation results.

Training set	Performance measures			
	MCC(%)	Specificity (%)	Accuracy (%)	Sensitivity (%)
1	42.47	94.89	95.90	94.78
2	32.85	95.01	95.91	93.89
3	53.17	93.67	96.01	94.12
4	42.37	94.56	95.67	92.67
5	52.27	95.85	96.69	95.89

5. Conclusion

This study proposed the novel method by utilizing a hybrid artificial neural network-based monarch butterfly optimization algorithm (HNNMBO) for liver tumor classification. In this study, the adaptively regularized kernel-based fuzzy C-means clustering algorithm and level enhanced segmentation algorithm are used for image segmentation. The hybrid DWT with partial Hadamard transform model is used for feature vector extraction thereby the liver tumor classification is carried out for the HNNMBO algorithm. The proposed work was implemented using MATLAB 2018a software in which images from the LiTS dataset. We have obtained 94%, 97.11%, 99.43%, 99.32%, and 99.87% segmentation performance by using Graph-based algorithm, ARKFCM with an enhanced level set, OT-LSEEIF-LDTP, ARKFCM with enhanced level set with the hybrid neural network with MBO and OT-LSEEIF-LDTP with the hybrid neural network with MBO methods. According to the classification examination, the proposed method provides 96.69% accuracy, 95.85% specificity, 95.89% sensitivity, 95.89% precision, 3.846% false-positive rate, 0.134% false-negative rate, 0.89% false discovery ratio, 95.38% negative predictive value, and 87.09% F-score values than other existing methods such as SVM, A-bLSTM, DL and HSA-HDL methods.

Acknowledgment

Authors would like to thank our mentor Late Dr. Basavaraj Amarapur for their continuous guidance and support.

Conflict of interest

There are no conflicts to declare.

Supporting information

Not applicable.

References

[1] X. Yang, H. Dou, R. Li, X. Wang, C. Bian, S. Li, D. Ni, and P. A. Heng, *International Conference on Medical Image*

Computing and Computer-Assisted Intervention, 2018, 497-505, doi: 10.1007/978-3-030-00937-3_57.

[2] S. Veerashetty, N. B. Patil, *Multimed. Tools. Appl.*, 2020, **79**, 9935–9955, doi: 10.1007/s11042-019-7345-6.

[3] B. A. Virupakshappa, *Health Technol.*, 2019, **9**, 701–713, doi: 10.1007/s12553-018-00288-y.

[4] R. L. Ambika, R. L. Biradar and V. Burkpalli, *Int. J. Comput. Appl.,* 2019, doi: 10.1080/1206212X.2019.1692442

[5] D. Zhang, J. Zhou, Y Yang and Q. Qin, *Symposium on Photonics and Optoelectronics,* 2012, 1-4, doi: 10.1109/SOPO.2012.6270911

[6] P Bilic, P F Christ, E Vorontsov, G Chlebus, Hao Chen, Q Dou, C W Fu, X Han, P. A. Heng, J. Hesser, S. Kadoury, T. Konopczynski, M. Le, C. Li, X. Li, J. Lipkovà, J. Lowengrub, H. Meine, J. H. Moltz, C. Pal, M. Piraud, X. Qi, J. Qi, M. Rempfler, K. Roth, A. Schenk, A. Sekuboyina, E. Vorontsov, P. Zhou, C. Hülsemeyer, M. Beetz, F. Ettlinger, F. Gruen, G. Kaissis, F. Lohöfer, R. Braren, J. Holch, F. Hofmann, W. Sommer, V Heinemann, C Jacobs, G E H Mamani, B van Ginneken, G Chartrand, A Tang, M Drozdzal, A Ben-Cohen, E Klang, M M. Amitai, E Konen, H Greenspan, J Moreau, A Hostettler, L Soler, R Vivanti, A Szeskin, N L Cohain, J Sosna, L Joskowicz, B H. Menze. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, H., Dou & Menze, B. H, *The liver tumor segmentation benchmark (lits)*, 2019.

[7] Q. Hu, J. Lee and Y. Luo, *Eng. Sci.*, 2019, **6**, 187-196, 10.30919/es8d507.

[8] T. Arun, Prasath D. Jithendra Reddy, *Int. J. Innov. Sci. Eng. Res.*, 2019, **6**, 29-38.

[9] S. K. Veerashetty, *Int. J. Innov. Sci. Eng. Res.*, 2021, **8**,141-148.

[10] A. B. A. Virupakshappa, *Lecture Notes in Electrical Engineering Springer,* 2018, **490**, doi: 10.1007/978-981-10-8354-9_21.

[11] Q. Hu, J. Y. Lee, and Y. Luo, *Eng. Sci.*, 2019, **6**, 12-21, doi: 10.30919/es8d507.

[12] B. Musunuri, S. Shetty, D. K. Shetty, M. K. Vanahalli, A. Pradhan, N. Naik and R. Paul, *Eng. Sci.*, 2021, **15**, 187-196, doi: 10.30919/es8d515.

[13] R. L. A. Biradar, *Health Technol.*, 2020, **10**, 231–247, doi: 10.1007/s12553-018-00289-x.

[14] R. Manjula Devi, and V. Seenivasagam, *Soft Comput.*, 2020, **24**, 18591-18598, doi: 10.1007/s00500-020-05094-1.

[15] H C Pan, Y. Li, N. Cai, L. Cheng, and S. Wang, *Ultrasonics*, 2021, 106271, doi: 10.1016/j.ultras.2020.106271.

[16] V. Sureshkumar, V. Chandrasekar, R. Venkatesan, and R. Krishna Prasad, *J. Amb. Intel. Hum. Comp.*, 2020, 1-8, doi: 10.1007/s12652-020-02107-7.

[17] Baneamoon, Saeed Mohammed, and Ali Salem Bin Sama, *Int. J. Comput. Sci. Mob. Computing.,* 2020, **9,** 50-55.

[18] S. Zhen, M. Cheng, Y. Tao, Yi. Wang, S. Juengpanich, Z. Jiang, Y. Jiang, *Front. Oncol.*, 2020, **10**, 680, 10.3389/fonc.2020.00680.

[19] M. Fujita, Y Rui, T. Hasegawa, S. Shimada, K. Arihiro, S. Hayashi, K. Maejima, *EBioMedicine,* 2020, **53**, 102659. doi: 10.1016/j.ebiom.2020.102659.

[20] Zeng, Y. and H. Dai, *J. Med. Imag. Health Infor.*, 2021, **11**, 337-344, doi: 10.1166/jmihi.2021.3295.

[21] M. Belge, M. E. Kilmer, and E. L. Miller, *IEEE T. Image Process*, 2000, **9**, 597-608, 10.1109/83.841937.

[22] C. Bezdek, Robert Ehrlich, and William Full, *Appl. Comput. Geosci.*, 1984, **10**, 191-203, doi: 10.1016/0098-3004(84)90020-7.

[23] Chandra, E., and K. Kanagalakshmi, *Int. J. Adv. Netw. Appl.*, 2011, **2**, 950-955.

[24] A. Elazab, C. Wang, F. Jia, J. Wu, G. Li, and Q. Hu, *Comput. Math. Method. Med.*, 2015, doi: 10.1155/2015/485495.

[25] Ibrahim, Rania, Ahmed Elbagoury, Mohamed S. Kamel, and F. Karray, *International Joint Conference on Neural Networks*, 2016, 2992-2999, 10.1109/IJCNN.2016.7727579.

[26] B. N. Li, B. Nan, J. Qin, R. Wang, M. Wang, and X. Li, *IEEE Access*, 2016, **4**, 4777-4788, doi: 10.1109/ACCESS.2016.2590440.

[27] L. Rada and K. Chen, *J. Algorithms Comput. Technol.*, 2013, **7**, 509-540, doi: 10.1260/1748-3018.7.4.509.

[28] A. Toifl, M. Quell, A. Hössinger, A. Babayan, S. Selberherr, and J. Weinbub, *International Conference on Simulation of Semiconductor Processes and Devices*, 2019, 1-4, doi: 10.1109/SISPAD.2019.8870443.

[29] L. A. Vese, and S. J. Osher, *J. Sci. Comput.*, 2003, **19**, 1, 553-572, doi: 10.1023/A:1025384832106.

[30] M. Rajawat, and D. S. Tomar, *Fifth International Conference on Communication Systems and Network Technologies*, 2015, 638-642, doi: 10.1109/CSNT.2015.245.

[31] S. Wang, G. Deng, and J. Hu, *Pattern Recogn.*, 2017, **61**, 447-458, doi: 10.1016/j.patcog.2016.08.017

[32] G. Goh, N. Cammarata, C. Voss, S. Carter, M. Petrov, L. Schubert, A. Radford, and C. Olah, *Distill,* 2021, **6**, e30, doi: 10.23915/distill.00030.

[33] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed and H Arshad, *Heliyon,* 2018, **4**, 11, doi: 10.1016/j.heliyon.2018.e00938.

[34] B. Musunuri, S. Shetty, D. K. Shetty, M K Vanahalli, A. Pradhan, N. Naik and R. Paul, *Eng. Sci.*, 2021, **15**, 187-196, doi: 10.30919/es8d515.

[35] S. Nandhini, and K. Ashok kumar, *Multimed. Tools. Appl.*, 2021, 1-28, doi: 10.1007/s11042-021-10599-4.

[36] H. Faris, I. Aljarah, and S. Mirjalili, *Appl. Intell.*, 2018, **48**,

445-464, doi: 10.1007/s10489-017-0967-3.

[37] B. S. Yıldız, A. R. Yıldız, E I Albak, H. Abderazek, S.M. Sait and S. Bureerat, *Mater. Test.*, 2020, **62**, 365-370, doi: 10.3139/120.111492.

[38] C. Mao, R. Lin, C. Xu, and Q. He, *IEEE Access*, 2017, **5**, 2187-2199, doi: 10.1109/ACCESS.2017.2654378.

[39] S. Zhen, Shi-hui, M. Cheng, Y-b Tao, Y Wang, S Juengpanich, Z Jiang, Y Jiang, *Front. Oncol.*, 2020, **10**, 680, doi: 10.3389/fonc.2020.00680.

[40] S. Randhawa, A. Alsadoon, P. W. C. Prasad, T. Al-Dala'in, A. Dawoud, and A. Alrubaie, *Multimed. Tools. Appl.*, 2021, **80**, 4729-4750, doi: 10.1007/s11042-020-09900-8.

[41] Y. Zeng, and H. Dai, *J. Med. Imag. Health Infor.,* 2021, **11**, 337-344, doi: 10.1166/jmihi.2021.3295.

Author Information

Deepak S Uplaonkar *received his Master degree in Computer Science & Engineering from Visvesvaraya Technological University Belagavi in the year 2013. He has more than ten years of teaching experience. He is currently carrying out his research in Poojya Doddappa College of Engineering Kalaburagi. He has published several papers in the standard international journals. His research interests include Pattern Matching, Computer Vision, Image Processing, Biomedical Image Processing, Data Science etc.*

Virupakshappa *received his Ph.D. from Visvesvaraya Technological University Belagavi in the year 2019 in the field of Biomedical Image Processing. Currently working as an Associate Professor in the department of Computer Science & Engineering in the Sharnbasva University Kalaburagi. He has published several papers in the international journals and conferences. He has over ten years of teaching experience. His research interest includes Digital Image Processing, Biomedical Image Processing, Pattern Matching, Machine Learning, Computer Vision etc.*

Nagabhushan Patil *currently working as Head of the Department of Electrical & Electronics Engineering in Poojya Doddappa College of Engineering Kalaburagi. He has received his Ph.D. in High Voltage Engineering in year 2017. He has thirty-five years of teaching experience. His area of research includes High voltage engineering, Analog Circuits, Digital Signal Processing, Digital Image Processing etc.*