



Classifying Hematoxylin and Eosin Images Using a Super-Resolution Segmentor and a Deep Ensemble Classifier

P. Sabitha* and G. Meeragandhi

Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai, Tamilnadu, India

*Corresponding Author: P. Sabitha. Email: sabithasanthosam11@gmail.com

Received: 16 July 2022; Accepted: 23 November 2022; Published: 23 June 2023

Abstract: Developing an automatic and credible diagnostic system to analyze the type, stage, and level of the liver cancer from Hematoxylin and Eosin (H&E) images is a very challenging and time-consuming endeavor, even for experienced pathologists, due to the non-uniform illumination and artifacts. Albeit several Machine Learning (ML) and Deep Learning (DL) approaches are employed to increase the performance of automatic liver cancer diagnostic systems, the classification accuracy of these systems still needs significant improvement to satisfy the real-time requirement of the diagnostic situations. In this work, we present a new Ensemble Classifier (hereafter called ECNet) to classify the H&E stained liver histopathology images effectively. The proposed model employs a Dropout Extreme Learning Machine (DrpXLM) and the Enhanced Convolutional Block Attention Modules (ECBAM) based residual network. ECNet applies Voting Mechanism (VM) to integrate the decisions of individual classifiers using the average of probabilities rule. Initially, the nuclei regions in the H&E stain are segmented through Super-resolution Convolutional Networks (SrCN), and then these regions are fed into the ensemble DL network for classification. The effectiveness of the proposed model is carefully studied on real-world datasets. The results of our meticulous experiments on the Kasturba Medical College (KMC) liver dataset reveal that the proposed ECNet significantly outperforms other existing classification networks with better accuracy, sensitivity, specificity, precision, and Jaccard Similarity Score (JSS) of 96.5%, 99.4%, 89.7%, 95.7%, and 95.2%, respectively. We obtain similar results from ECNet when applied to The Cancer Genome Atlas Liver Hepatocellular Carcinoma (TCGA-LIHC) dataset regarding accuracy (96.3%), sensitivity (97.5%), specificity (93.2%), precision (97.5%), and JSS (95.1%). More importantly, the proposed ECNet system consumes only 12.22 s for training and 1.24 s for testing. Also, we carry out the Wilcoxon statistical test to determine whether the ECNet provides a considerable improvement with respect to evaluation metrics or not. From extensive empirical analysis, we can conclude that our ECNet is the better liver cancer diagnostic model related to state-of-the-art classifiers.

Keywords: Convolutional block attention modules; dropout ELM; ensemble classifier; liver cancer; segmentation; voting mechanism



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

Nowadays, scientists and researchers used the ML and DL models in several applications including agriculture, environment, text sentiment analyses, cyber security, and medicine [1]. Early detection of liver tumors is one of the hot research subjects in the literature since it is the sixth most common malignancy and the fourth-highest leading cause of fatalities around the world [2]. It is expected that, by 2025, around one million lives will be suffered from liver cancer [3]. Many sociodemographic features such as sex, age, and geographical region are strongly associated with liver cancer. The malignant cells in the liver can enchain together to create cancer and then spread to other tissues and organs (e.g., bones, lungs, etc.). Hepatocellular Carcinoma (HCC) is an extremely aggressive primary cancer and accounts for 90% of the cases worldwide [4]. Hence, early diagnosis of HCC is certainly imperative and requires to be carried out accurately.

Manual HCC diagnosis and segmentation of the nuclei regions in H&E stained histopathology images is a time-consuming and challenging process. Moreover, the accuracy of the manual diagnostic system is dependent on the complexity of the disease and the pathologist's diagnostic knowledge [5], which eventually leads to an imprecise assessment. As morphological attributes of H&E stains are very complex the study of these images is very difficult and it is still in the exploratory phase in the application of clinical pathology. At the same time, these challenging factors motivated several investigators to develop novel diagnostic triage and testing strategies to support the early detection of liver cancer which can identify the tumor cells with greater accuracy and consequently improve clinical outcomes. Furthermore, the computer-aided diagnosis will be far faster than the manual screening.

DL-based HCC diagnostic systems have made noteworthy strides in the past decades; hence, several effective automatic diagnostic models have been established. Current advancements in DL approaches provide new opportunities for HCC identification and classification which have gained attention from research, academic and fiscal societies. DL models generally include various phases such as preprocessing, segmentation, feature extraction, and classification. The apposite preprocessing technique reduces the deviations in image quality, for example, denoising to minimize noise and artifacts in the image, enhancement technique to improve the contrast between background and region of interest [6], spatial filtering to enhance edges and boundaries by eliminating blur [7], and color normalization to reduce color variation in the image [8].

In HCC diagnosis, segmentation is a vital process that extracts the important tissues (e.g., nuclei, stroma, and lymphocyte) that have the required attributes to identify cancer cells. When a DL approach is employed to support liver cancer detection, efficient feature engineering can be implemented through medical oncologists, which makes it viable to realize improved enactment [9]. Apt attributes can be extracted by applying attribute selection approaches according to the pathologist's knowledge, or other techniques. In classification, DL algorithms are used to categorize the segmented histopathology images based on their features. Conversely, several DL models may hamper the effectiveness of the collaboration between physicians and the system owing to the perplexing decision-making procedure [10,11]. Hence, increasing the efficiency of liver cancer detection systems with greater comprehensiveness (i.e., higher prediction accuracy and generalization against fluctuating data) is inevitable for automatic cancer detection models [12].

Of late, Deep Ensemble Learners (DEL) have been proposed in the literature that integrates more than one DL approach to provide improved classification results and better generalization performance [13]. Related to the individual DL classifiers, an ensemble model accomplished higher classification accuracy. DEL frequently chooses an optimum set of individual classifiers and then integrates them by means of a precise fusion technique such as stacking, support function fusion, and majority voting [14]. Therefore, the decision on selecting individual classifiers and assimilating them is critical. In order to achieve the best enactment, the selected classifiers should have optimal enactments as well as sufficient diversity [15].

Thus, the performance of these systems still needs substantial enhancement, and then only they can fulfill the demands of real-time diagnostic scenarios. In this work, we propose a DEL model to detect and classify liver cancer cells from H&E histopathological images with better accuracy. The major contributions of our work are four-fold.

1. A super-resolution convolutional network is developed to segment various edges of nuclei in the H&E liver image.
2. An efficient DEL classifier is developed with two base learners using a dropout extreme learning machine and enhanced convolutional block attention module-based residual networks. The proposed ECNet employs a voting mechanism to integrate the decisions from base learners.
3. The proposed ECNet is implemented using two real-world liver cancer datasets including KMC and TCGA-LIHC and its performance is compared with some state-of-the-art classifiers in terms of selected performance metrics.

The rest of this paper is arranged as follows: Section 2 presents the basic concepts of DL classifiers. Section 3 analyses the relevant works about DL-based liver cancer classification techniques. In Section 4, we discuss the proposed ECNet in detail. Then the experimental setup is discussed in Section 5. The numerical fallouts are given in Section 6. Section 7 concludes this work.

2 Preliminaries of DL Classifiers

Generally, DL classifiers include five basic layers, viz. convolution, activation, pooling, fully-connected (dense), and output layers. The convolutional layer aims to learn feature maps (also called channels) from the dataset by applying different kernels (filters or feature detectors), by which their spatial relationship can be retained [16]. In this architecture, every pixel of a channel is associated with a region of adjacent pixels in the preceding unit. The new channel can be achieved by convolving the input with a trained filter and then performing a pixel-wise nonlinear transfer operation on the convolved fallouts. Each spatial location of the input disseminates the filter to create a channel. The feature value ($\eta_{p,q,n}^l$) at position (p, q) in the n^{th} channel of the l^{th} layer is defined by Eq. (1) [17].

$$\eta_{p,q,n}^l = \omega_n^{lT} I_{p,q}^l + \beta_n^l \quad (1)$$

where ω_n^l is the weight vector and β_n^l is the bias term of the n^{th} kernel of the l^{th} layer, correspondingly. The term $I_{p,q}^l$ is the input patch focused at (p, q) of the l^{th} layer. $\eta_{p,q,n}^l$ represents the shared channel which is generated by the kernel ω_n^l . This weight disseminating technique can reduce the computational overhead and make the model simpler to train. The activation (transfer) operation adds nonlinearities to the classifier which is required for multi-layer classifiers to identify nonlinear attributes. Consider $a(\cdot)$ is the nonlinear transfer operation. The activation value ($a_{p,q,n}^l$) of convoluted attribute $\eta_{p,q,n}^l$ can be calculated using Eq. (2).

$$a_{p,q,n}^l = a\left(\eta_{p,q,n}^l\right) \quad (2)$$

In general, hyperbolic tangent, sigmoid, and Rectified Linear Unit (ReLU) [16,17] are used as transfer functions. However, ReLU is one of the most widely used non-saturated transfer operations and it is defined as given in Eq. (3).

$$a_{p,q,n}^l = \max\left(\eta_{p,q,n}^l\right) \quad (3)$$

The subsampling (pooling) operation is employed to decrease the size of the attribute space obtained from the convolutional function. The pooling layer is generally located between two convolution layers.

Every channel of a subsampling layer is coupled to its respective channel of the previous convolution unit. The pooling operation ($pool(\cdot)$) of a particular channel $a_{p,q,n}^l$ is defined by Eq. (4).

$$\rho_{p,q,n}^l = pool\left(a_{r,s,n}^l\right), \quad \forall(r, s) \in R_{pq} \quad (4)$$

where R_{pq} is a local region about the point (p, q) . After stacking some convolution and subsampling blocks, DL networks consist of one (or multiple) dense layers which target to achieve higher-level attributes. These layers consider all nodes in the preceding unit and link them to each node of the current layer to produce global semantic data. However, this operation is not always required as it can be substituted by a 1×1 convolutional operation. The final layer of DL classifiers is an output layer. The softmax operator is frequently employed for classification. Suppose we have k number of data, the anticipated input-output correlations is defined by $\{I(k), J(k); k \in [1, 2, 3 \dots k]\}$, in which $I(k)$ is the k^{th} input data, $J(k)$ is its equivalent target class and $O(k)$ is the output of the classifier. The loss function of the classifier can be computed using Eq. (5).

$$L = \frac{1}{K} \sum_{k=1}^K l(\emptyset; I(k), O(k)) \quad (5)$$

Training DL classifiers is a global optimization problem. We can calculate the optimal system variables by minimizing the loss function.

3 Related Works

To date, diagnosis of HCC from histopathology images using DL techniques is an emerging field in the medical industry. With the advent of cutting-edge diagnostic imaging and DL techniques, early diagnosis of HCC is possible. Recently, Convolutional Neural Networks (CNN) have become the advanced tool for classifying medical images. Huang et al. proposed the Densely connected convolutional Networks (DenseNet) by adapting the skip-connection of the ResNet architecture and connecting all convolution layers [18]. For every layer in the DenseNet, the input denotes the channels of all the previous units, while its output channels are used as inputs to the subsequent units. Szegedy et al. presented a sophisticated model with an inception module to increase the width and depth of the system while keeping the same level of processing overhead [19]. Ioffe et al. proposed an improved inception network (Inception-V2) by employing batch normalization after performing a convolution operation [20].

Szegedy et al. refined the previous version of the inception network to develop the InceptionV3 module by implementing the concept of extra factorization to provide better performance [21]. But, the Inception-V3 network generates several network variables which leads to an overfitting issue and consequently increases the computational complexity. Later, Szegedy et al. along with a Google research team developed a classification model called I-ResNetV2 by integrating the concepts of InceptionV3 and ResNet models [22]. The resultant network replaces the kernel concatenation of inception with skip-connection to integrate the paybacks of the two models (i.e., making wider and deeper) while maintaining the same degree of system complexity. Ferreira et al. proved that a pre-trained I-ResNetV2 could be employed for identifying tumor cells in medical scans using transfer learning [23]. To classify medical images, Alom et al. developed the Inception Recurrent Residual Convolutional Neural Network (IRRCNN) [24]. The proposed model exploits a recurrent CNN unit connected with the inception unit. Togcar et al. introduced the BreastNet model for performing breast cancer identification [25]. The BreastNet model exploits the bilinear upsampling module to excerpt attributes from three different depths. Raza et al. proposed a unified model called MicroNet [26]. This model integrates an extra layer in the downsampling path to keep the feeble nuclei attributes which may be skipped in the max-pooling function.

Woo et al. proposed an efficient attention unit for the ResNet model, called CBAM [27]. This unit includes two consecutive components including a channel attention unit and a spatial attention unit. The intermediary channel is adaptively optimized by CBAM at each convolution layer of ResNet. The authors demonstrate that the total computational complexity of CBAM is relatively small regarding the number of variables and computation. This encourages us to implement an improved version of CBAM in our proposed classification model to classify the liver histopathological images. Dogantekin et al. proposed a classification model using a convolutional Extreme Learning Machine (ELM) to decrease the processing and storage complexity without mortifying the classification performance in classifying liver cancer [28]. This model uses five convolutional and two dense layers. The attributes of the final layer are fed as a source to the ELM classification algorithm. However, this algorithm provides better clinical outcomes when adequate hidden layers are employed. Aatresh et al. proposed an effective and reliable DL approach for automated classification of liver cancer from H&E stained liver histopathology data, called LiverNet [29]. This model uses CBAM with spatial pooling blocks for efficient liver cancer identification. Though the classification enactment of the liver cancer diagnostic system is increased through the abovementioned models, the variables and other processing overhead are still very high for simple applications. Most of the classification algorithms could effectively categorize the medical scan into benign and malicious. Even though, the segmentation algorithms used in these models provide acceptable results unraveling overlapped nuclei remains a perplexing task. To the best of our knowledge, our ECNet is among the first deep ensemble learner to develop a robust classification approach for HCC detection.

4 Proposed DEL-Based Liver Cancer Diagnostic System

The proposed deep ensemble diagnostic system integrates a deep segmentation algorithm to segment various contours of nuclei in H&E images and a deep ensemble classifier to detect and classify liver cancer cells from H&E stained histopathology scans. In the ECNet model, the nuclei regions in the liver image are first extracted through the super-resolution networks. Then, the segmented nuclei regions are fed into a DEL for classification. This deep ensemble classifier contains two different classifiers including the DrpXLM classifier and ECBAM-based ResNet. Fig. 1 illustrates the architecture of the proposed ECNet.

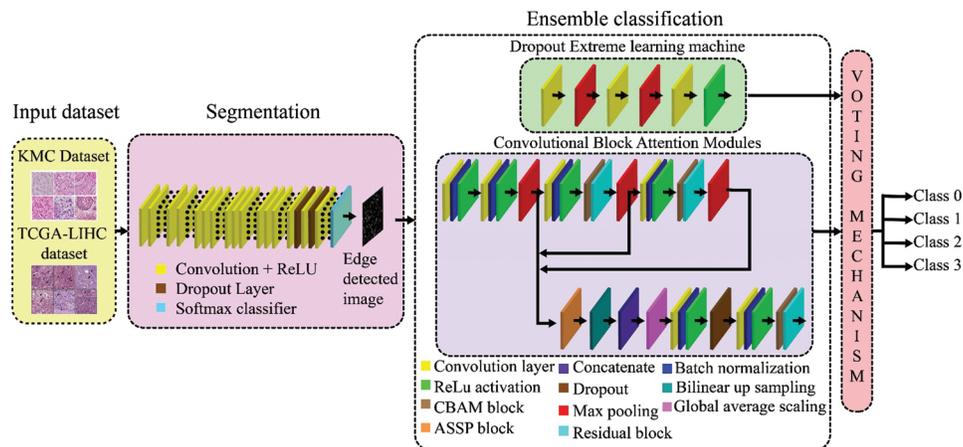


Figure 1: Architecture of ECNet for liver tumor segmentation and classification

4.1 Nuclei Segmentation Using SrCN

Nuclei segmentation is a vital process in the automatic liver cancer diagnostic system since it can extract edge information of nuclei size and shapes which possess vital attributes of liver tumor. Thus, it makes the

tumor detection process easier and minimizes its processing overhead significantly. Our SrCN is a high resolution-preserving DL approach to produce very accurate segmentation. In this network, the RGB (histopathology) images are given as input for extracting the nuclei. The SrCN learns higher-level attributes by relating the source scans to its analogous ground-truth binary masks with slight resolution loss. The super-resolution channels of each input pixel are preserved by eliminating each pooling layer in the SrCN network. The SrCN segmentor consists of 16 convolution layers. But, the last three Fully-Connected Layers (FCL) are replaced by the convolution layers to enable the system to create dense prediction maps with an equal dimension as the source scans. The initial layers of SrCN learn local statistics and high-level feature representations whereas the last three layers learn delicate and low-level features of nuclei edges.

In this work, we apply the color normalization technique to minimize inter-color variation in a set of input scans. We achieve color normalization by transforming color from a target scan and also improving the contrast for fade scans to some extent. In this model, boundary identification is carried out according to the measured local Standard Deviation (SD) value in an $s \times s$ window (mask) about every pixel in the image. By applying the SD value, the SrCN segmentor is less sensitive to noise. The initial phase of boundary finding is to consider only 'r' space statistics of the color normalized scan, as it generally holds foreground statistics (i.e., nuclei), which are perceived in the entire database. When transforming the RGB scan to 'r' space information numerous stains are spontaneously unglued from the image. Hence, this proposed model did not use any stain isolation technique. The 'r' space scan is then processed with a non-linear kernel, which is the inverse logarithm of the digital Gaussian kernel in a 3×3 mask. This kernel decreases noise at a certain level in H&E scans. The user selects n number of input images with different microscopic zooms (such as $10\times$, $30\times$, or $40\times$). Then, 'n' is multiplied by 0.15 and kept in a parameter 's'. Accordingly, for $10\times$ scans, $s = 2$, for $30\times$ scans, $s = 5$, and for $40\times$ scans $s = 6$.

Algorithm 1: Edge detection algorithm

```

1 I =imread(color normalized scan)
2 [r,g,b] = imsplit(I);
3 nonlinearFilter = 0.5 [1/log2 (GaussFilter(3,3))]
4 I2 = nonlinearFilter(I)
5 n = 20 or 40
6 def bounddetect (I)
7   Sh = image.shape
8   con_img = np.zeros(Sh)
9   for x in range (Sh[0] - 12)
10    for x in range (Sh[0] - 12)
11      tot = 0
12      s=n*0.15
13      st = np. sd(image [x:(x+s), z:(z+s)])
14      for i in range (3)
15        for i in range (3)
16          tot = tot +(image[x+i, z+j] *(st)
17      con_img [x,z] = tot

```

(Continued)

Algorithm 1 (continued)

```

18         return con_img
19 I3 = bounddetect(I2)
20 I3 = (abs(I3))
21 I3 = 255 - I3
22 matplotlib.pyplot.imshow(I3, cmap = 'Greys')
23 matplotlib.pyplot.imshow ()

```

The filtered greyscale scan is further convolved with a 3×3 window with variable factors. Initially, we consider the SD of ' $s \times s$ ' pixels about the first pixel in the scan, and the equal SD value is assigned in all nine variables of a 3×3 window. Next, the intensity of pixels (where a 3×3 window is being employed) is multiplied by the window factor and then we aggregate those results in a variable 'tot' and substitute the initial intensity of each pixel with 'tot', as illustrated in Algorithm 1. Subsequently, this 3×3 mask (window) is moved right side of the scan by one step and the process of allocating intensity values to each pixel of the scan is repeated. Fig. 2 illustrates various steps in our SrCN segmentor. The boundary of the segmented nuclei is resized into a constant dimension of 256×256 pixels through bi-linear interpolation for all learning, validation, and testing data. Then, our deep ensemble learner with DrpXLM and ECBAM-based ResNet classification algorithms is used for the segmented histopathology images for classification. The ensemble classifier can exploit the strong points of the base learners and can mitigate their febleness, ultimately realizing a greater classification performance than any individual classifiers.

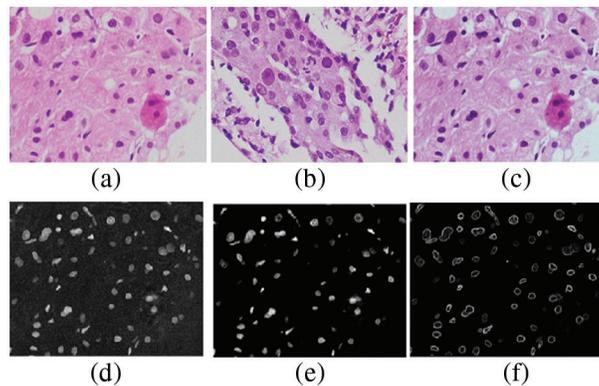


Figure 2: Edge detection (a) source scan (b) target scan (c) color normalized scan (d) color normalized scan for taking 'r' space information (e) 'r' space information after using non-linear filter (f) edge detected scan

4.2 Deep Ensemble Classification

Due to the diversity of disease types and the heterogeneity of databases, individual deep learners are likely to limit the performance of disease detection and classification models. In line with this issue, we propose a DEL model to integrate simple and effective classifiers. Our ECNet employs two classifiers including DrpXLM and ECBAM-based ResNet classifiers. ECNet applies a voting mechanism to integrate the decisions of individual classifiers by applying the average of probabilities rule. Implementation of this integrated ensemble model with a voting mechanism can provide both better accuracy and diversity in the classification system.

4.2.1 Dropout Extreme Learning Machine

An extreme learning machine is an effective deep learner model for training Single hidden Layer Feed-forward neural Networks (SLFNs) with high speed and better generalization aptitude. In this approach, the weights and bias terms of the input and hidden layers are arbitrarily selected, whereas the hyperparameters of the output layer are systematically derived by calculating a generalized inverse matrix. Conversely, there are two important issues in ELM including the selection of structural design and the prediction variability. To address these issues, we apply the dropout method in the ELM classification model. The rudimentary concept of the dropout mechanism is to arbitrarily discard connections along with corresponding units from deep learners during the learning process, which averts their co-adaptation due to the overfitting problem. SLFN with h hidden neurons is defined using Eq. (6).

$$f_h(x) = \sum_{i=1}^h A_i(x_i, w_i, \beta_i) O_i \quad (6)$$

where A_i represents the activation operation; $w_i \in R^d$ is the input weight vector; $\beta_i \in R$ is the bias term, and $O_i \in R^m$ is the output weight vector of the i^{th} hidden neuron. In extreme learning, w_i and β_i are arbitrarily selected, whereas O_i is computed logically. The additive nodes and radial basis function neurons are frequently employed as hidden neurons. If these neurons are additive with transfer function $a(\cdot)$, then A_i is expressed as given in Eq. (7).

$$A_i(x, w_i, \beta_i) = a(w_i x + \beta_i) \quad (7)$$

If the hidden neurons are radial basis function nodes with transfer function $a(\cdot)$, then A_i is expressed as given in Eq. (8).

$$A_i(x, w_i, \beta_i) = a(\beta_i \|x - w_i\|) \quad (8)$$

Given a training dataset $T = \{(x_i, y_i) | x_i \in R^d, y_i \in R^k, i = 1, 2\}$, where x_i is an input vector with the size of $d \times 1$ and y_i is a target vector with the size of $k \times 1$. Since $A_i(x, w_i, \beta_i)$ in Eq. (6) defines a similar transfer operation, we assume $A_i(x_i, w_i, \beta_i) = A(x, w, \beta)$, and replace x_i and y_i by x and $f_h(x)$ in Eq. (6), correspondingly, to get the following Eq. (9).

$$y_i = \sum_{j=1}^h A_i(x_i, w_j, \beta_j) O_j \quad (9)$$

where $i = 1, 2, 3 \dots h$. Eq. (9) can be inscribed in a more compact form as given in Eq. (10).

$$GO = Y \quad (10)$$

where

$$G = \begin{bmatrix} A(x_1, w_1, \beta_1) & \dots & A(x_1, w_h, \beta_h) \\ \dots & \dots & \dots \\ A(x_n, w_1, \beta_1) & \dots & A(x_n, w_h, \beta_h) \end{bmatrix}, \quad O = (O_1^T, O_2^T, \dots, O_h^T)^T, \quad \text{and} \quad Y = (Y_1^T, Y_2^T, \dots, Y_h^T)^T$$

G is the output matrix of the hidden layer, where i^{th} column of G is the i^{th} output vector of hidden nodes relating to inputs x_1, x_2, \dots, x_n , and the i^{th} row of G is the output vector of the hidden layer relating to input x_i . If the number of hidden neurons is the same as the number of individual learning data samples, the matrix G is square and invertible, and SLFN can estimate these learning datasets with zero error. However, in most cases, the number of hidden neurons is considerably smaller than the number of learning datasets. Consequently, G is a non-square matrix and we cannot find an optimal solution to the system. However, we can determine its optimal (smallest norm least-squares) solution by resolving the objective function given in Eq. (11).

$$\min_{\beta} \|GO - Y\| \quad (11)$$

The optimal solution of Eq. (11) can be calculated by Eq. (12).

$$\hat{O} = G^\dagger Y \quad (12)$$

where $G^\dagger = (GG^T)^{-1}G$ is the Moore-Penrose generalized inverse of matrix G . In this work, we implement the dropout method in the original ELM classifier to address the overfitting problem. The dropout mechanism arbitrarily discards units in consort with their links from deep models during learning. This technique averts the co-adaptation issue due to the overfitting problem in the dataset. Given an input vector $I = \{I_1, I_2, \dots, I_n\}$, the output of linear node $n(I)$ is defined using Eq. (13).

$$\gamma(I) = \sum_{i=1}^n w_i I_i \quad (13)$$

If we drop I_i ($1 \leq i \leq n$) with a uniform distribution or drop I_i ($1 \leq i \leq n$) with a likelihood of $\frac{1}{2}$. Consequently, we get 2^n subnets with the empty network. The average of outputs of the 2^n subnets is calculated using Eq. (14).

$$O_{avg}(\gamma) = \frac{1}{2^n} \sum_{\mathbb{N}} \gamma(\mathbb{N}, I) \quad (14)$$

where \mathbb{N} is the set of all subnets. The term α_i ($1 \leq i \leq n$) represents n Bernoulli arbitrary variables and $r_i = R(\alpha_i = 1)$, $s_i = S(\alpha_i = 0)$. If we drop I_i ($1 \leq i \leq n$) or w_i ($1 \leq i \leq n$) with a likelihood of s_i then, the output of the linear node is calculated by Eq. (15).

$$\gamma(I) = \sum_{i=1}^n w_i \alpha_i I_i \quad (15)$$

Now, we can calculate the mathematical expectation of the output of linear neurons using Eq. (16).

$$E(\gamma) = \sum_{i=1}^n w_i E(\alpha_i) I_i = \sum_{i=1}^n w_i r_i I_i \quad (16)$$

If $r_1 = r_2 = \dots = r_n = r$, then formula (16) becomes the following Eq. (17).

$$E(\gamma) = \sum_{i=1}^n w_i E(\alpha) I_i = \sum_{i=1}^n w_i r I_i \quad (17)$$

If there is a bias term β in the linear node, then the output of the linear node can be defined as Eq. (18).

$$\gamma(I) = \sum_{i=1}^n w_i r I_i + \beta r_\beta \quad (18)$$

In the above equation, $r_\beta = P(\alpha_\beta = 1)$.

4.2.2 Enhanced CBAM-Based ResNet Classifier

Woo et al. introduced the idea of CBAM [27]. The authors exploit sequential channel attention and spatial attention units to provide attention maps that can be multiplied against the input channel. They also proved that the CBAM can enhance prediction accuracy when it is incorporated into advanced classifiers. The CBAM-based deep classifiers pay attention to the region of interest in the network. Given an intermediate channel $\mathcal{F} \in \mathbb{R}^{W \times D \times C}$ is given as an input to the CBAM at an intermediary point in terms of depth (D), width (W), and cardinality (C) in the encoder pipeline. CBAM successively deduces a 1D channel attention map $\mathcal{A}_c \in \mathbb{R}^{1 \times 1 \times C}$ and a 2D spatial attention map $\mathcal{A}_s \in \mathbb{R}^{W \times D \times 1}$. The entire attention task can be defined as given in (19).

$$\begin{cases} \mathcal{F}' = \mathcal{A}_c(\mathcal{F}) \otimes F \\ \mathcal{F}'' = \mathcal{A}_c(\mathcal{F}') \otimes \mathcal{F}' \end{cases} \quad (19)$$

where \otimes represents component-wise multiplication. \mathcal{F}' is the intermediary result following the component-wise multiplication between the $\mathcal{A}_c \in \mathbb{R}^{1 \times 1 \times C}$ and \mathcal{F} . The term \mathcal{F}'' is the ultimate result of the attention unit, and also the product of component-wise multiplication between $\mathcal{A}_s \in \mathbb{R}^{W \times D \times 1}$ and \mathcal{F}' . We create \mathcal{A}_c using the inter-channel correlation between attributes. The channel attention block utilizes both max-pooling and average-pooling results with a shared dense network. First, we integrate spatial statistics of a channel by applying both average pooling and max-pooling functions. We introduce two spatial context identifiers \mathcal{F}_{avg}^c and \mathcal{F}_{max}^c to represent average- and max-pooling attributes, correspondingly. Both identifiers are then disseminated to a dense layer to generate $\mathcal{A}_c \in \mathbb{R}^{1 \times 1 \times C}$. The shared network contains FCL with one hidden unit. To minimize parameter cost, the size of the hidden activation is fixed to $\mathbb{R}^{1 \times 1 \times C/r}$, where r is the reduction factor. Once the dense layer is used for every identifier, we combine the attribute vectors through component-wise addition. In brief, the channel attention is determined by Eqs. (20) and (21).

$$\mathcal{A}_c(\mathcal{F}) = \rho(FCL(AvgPool(\mathcal{F})) + FCL(MaxPool(\mathcal{F}))) \quad (20)$$

$$\mathcal{A}_c(\mathcal{F}) = \rho\left(\omega_1\left(\omega_0\left(\mathcal{F}_{avg}^c\right)\right) + \omega_1\left(\omega_0\left(\mathcal{F}_{max}^c\right)\right)\right) \quad (21)$$

where ρ represents the sigmoid function, $\omega_1 \in \mathbb{R}^{c/r \times c}$ and $\omega_0 \in \mathbb{R}^{c \times c/r}$. The FCL parameters (ω_1 and ω_0), are disseminated for both inputs and the ReLU activation function. We create a spatial attention map using the inter-spatial correlation of attributes. On the connected attribute identifier, we implement a convolutional unit to produce $\mathcal{A}_s(\mathcal{F}) \in \mathbb{R}^{W \times D \times 1}$ which deduces where to emphasize or quash. The residual unit employed in our ECNet model is similar to the residual unit discussed in [20] and has been implemented in [25]. The only difference being the kernels of the first convolutional unit in our residual unit is decreased by a ratio of 4 related to the kernel employed in the residual unit discussed in [25]. This not only decreased the number of system variables essential for classification but also increased the attributes gained from the input.

An atrous spatial pyramid (ASP) pooling unit can effectively gain multi-scale attributes from a channel. Considering its efficacy, we implement an identical ASP pooling block in our classifier. Atrous (dilated) convolutional operation can be employed to increase the receptive arena dimension without increasing the number of variables used. Consider an input 2D signal X convolved with a 2D kernel ω using dilated convolutional operation. This operation is defined as given in Eq. (22).

$$Y[i,j] = \sum_h \sum_n X[i+r.m, j+r.n] \omega[h][n] \quad (22)$$

where r denotes the rate at which the input is sampled (i.e., dilation rate). The impact of dilation increasing the size of the receptive field of the filter is equal to appending $(r-1)$ zeros in between the filter components. Hence, for $r=2$, a 3×3 filter will have a receptive field size corresponding to a 5×5 filter but with only nine variables. A similar idea can be applied for higher values of $r=4, 5, 6$, etc. The ASP pooling unit employed in our model considers a channel as input and perform the following convolutional operations simultaneously: (i) 1×1 ; (ii) 3×3 with $r=2$; (iii) 3×3 with $r=3$; (iv) 3×3 with $r=6$ (v) 3×3 with $r=8$; and (vi) global average subsampling. Each convolutional function in each branch is performed with a batch-normalization and a ReLU transfer function. In this model, we have employed depthwise separable convolutions to decrease the number of variables and make the model more effective. This operation needs only $\frac{1}{8}$ th the number of variables and only $\frac{1}{9}$ th the number of iterations. The expression for depthwise separable convolutions is designated by considering an input scan I of dimension $D_f \times D_f \times M$ and a kernel w of

dimension $D_k \times D_k \times N$. The depthwise separable convolution can be performed in 2 steps: (i) depthwise convolution; and (ii) pixel-wise convolution. In the first step, the image (I) is convolved with ω_D kernel with the dimension of $D_k \times D_k \times 1$.

$$I_D = I \otimes \omega_D \quad (23)$$

where the dimension of I_D is $D_p \times D_p \times M$. After performing this function, the pixel-wise convolutional function employs a $1 \times 1 \times M$ filter convolved with the intermediary result for each size N of the basic filter ω . This can be defined in Eq. (24).

$$I_P = I_D \otimes \omega_P \quad (24)$$

where ω_P is the filter of dimension $N \times 1 \times 1 \times M$ and I_P is the ultimate output from the convolutional layer. The results obtained from convolutional and subsampling operations are combined and fed into a 1×1 convolutional unit to keep the same kernel dimension as that of the input image. This output of the convolutional layer is fed into batch normalization and ReLU transfer units before they are moved into the bilinear upsampling unit. In our classifier, we perform the ASP pooling operation after performing every max-pooling operation in the encoder block. The max-pooled output comprises higher-level attributes at various extents and scales. ASP pooling units are employed in this model to extract this data before it is upsampled and handled by the hyper-column method.

4.2.3 Voting Mechanism

The estimation of each contributing base learner in the deep ensemble method may be preserved as a vote for a particular class, i.e., normal (class 0) or malevolent (class 1, class 2, and class 3). It exploits two individual classification algorithms (i.e., DrpXLM and ECBAM-based ResNet) and implements an integration rule for making decisions. In the ECNet model, we use an average likelihoods technique to make a decision. In this model, the class label is selected based on the maximum value of the average of likelihoods. Let u represents the number of base learners $C = \{C_1, C_2, \dots, C_u\}$ with c classes $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_c\}$. In our experiment, we select the values $c = 4$ and $m = 2$. A classifier $C_i: R^n \rightarrow [0, 1]^c$ accepts an input sample $x_i \in R^n$ and provides the output as a vector $\delta_{ci}(\varphi_1|x), \delta_{ci}(\varphi_2|x), \dots, \delta_{ci}(\varphi_c|x)$, where $\delta_{ci}(\varphi_k|x)$ denotes the likelihood assigned by C_i that input sample x fits into class φ_k . The term AL_k is the average of likelihoods assigned by the classifiers for each class. It can be computed using Eq. (25).

$$AL_k = \frac{1}{m} \sum_{i=1}^m \delta_{ci}(\varphi_k|x) \quad (25)$$

Let $AL = [al_1, al_2, \dots, al_c]$ is the set of the average of likelihoods for c classes and x is assigned to the weight φ_c if AL_k is having a maximum value in the set AL .

5 Experimental Setup

The proposed ECNet classifier is realized using MATLAB 2017b Deep Learning Toolbox. The experiments are carried out on Intel® Core™ i7-6850K CPU with 3.360 GHz 16 GB RAM and a CUDA-enabled GPU of NVIDIA GeForce GTX 1080. The effectiveness of our model is demonstrated using two real-world liver cancer datasets including KMC and TCGA-LIHC and its performance is compared with some state-of-the-art classification models in terms of evaluation metrics such as accuracy, sensitivity, specificity, precision, and JSS.

5.1 Dataset Preparation

To demonstrate the effectiveness of ECNet, we use a corpus of labeled liver tissue images from two datasets: KMC and TCGA-LIHC [26]. The KMC dataset contains 257 recently collected original liver images by pathologists from different patients. Table 1 shows the statistical report of each instance in this dataset. We apply normalization and resizing methods as data reprocessing methods to all datasets. The databases have been normalized in the range of $[-1, +1]$ before segmentation. The normalization improves the consistency among input scans, and it delivers steady convergence. The images are resized to 224×224 before giving in to the classifier using bilinear interpolation. To realize a more precise evaluation of a model's enactment, the k -fold cross-validation (CV) is used. ECNet assigns k equals 10. In a 10-fold CV, the whole database is divided into 10 parts. For each folding, one part of the dataset is used for testing and the other part is combined for training. Then, we compute the average value of results across all ten autonomous trials. Indeed, a separate set of data samples for testing and validation are not presented in the database. Therefore, we have deliberately split the existing dataset into 70% for learning, 20% for testing, and 10% for validation. Furthermore, since we used 10-folds CV, the total data samples are split into 10 parts (each of 10%). Now, one fold (10%) is applied for testing, while the residual samples (90%) are split for validation and training. The utilization of 10-fold CV guarantees that each slide in the data gets to be in a test exactly once

Table 1: Original data distribution of the datasets (the bold values refer to the total number)

Class type	KMC dataset				TCGA-LIHC			
	Training	Validation	Testing	Total	Training	Validation	Testing	Total
Class 0	49	7	14	70	49	7	14	70
Class 1	56	8	16	80	45	6	13	64
Class 2	58	8	17	83	48	7	14	68
Class 3	17	3	4	24	49	7	14	70
Total	180	26	51	257	191	27	54	272

5.2 Evaluation Measures

To evaluate the effectiveness of ECNet using the deep ensemble classifiers, we used five significant evaluation metrics including classification accuracy, sensitivity, specificity, precision, and JSS. These performance measures are required to be higher to improve the effectiveness of the proposed model. The utility is computed in terms of classification accuracy as given in Eq. (26).

$$\text{Accuracy (ACC)} = \frac{TN + TP}{TN + TP + FN + FP} \quad (26)$$

where true positive (TP) signifies the number of images correctly classified as malignant; false negative (FN) represents the number of malignant images incorrectly recognized as normal. The term true negative (TN) signifies the number of images correctly classified as normal and false positive (FP) describes the number of normal images mistakenly classified as malignant ones. Sensitivity is the rate of disease detection and specificity signifies the false alarm rate.

$$\text{Sensitivity (SEN)} = \frac{TP}{TP + FN} \quad (27)$$

$$\text{Specificity (SPE)} = \frac{FP}{TN + FP} \quad (28)$$

Precision identifies the abnormal data samples that accurately belong to the malevolent class. It is computed using Eq. (29).

$$\text{Precision (PRE)} = \frac{TP}{TP + FP} \quad (29)$$

JSS estimate delivers the similarity between the ground truth and the classified data. It is calculated using Eq. (30).

$$\text{Jaccard similarity score (JSS)} = \frac{TP}{TP + FN + FP} \quad (30)$$

The Wilcoxon statistical test is conducted to determine whether the ECNet classifier provides a remarkable enhancement compared to other classifiers or not at a level of 5% statistical importance [28]. If the outputs of this test $\rho < 5\%$ signify that there is a significant difference at this level of significance. Otherwise there is no noteworthy difference between the related outcomes.

6 Results and Discussion

The performance of our ECNet classifier is evaluated by comparing the numerical results with that of 9 state-of-the-art classifiers, such as ResNet [20], DenseNet [18], I_ResNetV2 [23], IRRCN [24], BreastNet [25], MicroNet [26], CBAM [27], ELM [30], and LiverNet [29]. All these networks were applied, analyzed, and compared by applying the same learning, testing, and validation data samples through MATLAB 2017b DL Toolbox. The proposed classifier network is applied to the KMC and TCGA-LIHC datasets. The comprehensive results achieved by the proposed model are given in Table 2. By applying our model on KMC dataset, we can find that the ResNet model has achieved nominal enactment with 86.4% accuracy, 98.1% sensitivity, 68% specificity, 82.8% precision, and 76.5% JSS by introducing the trainable layers. The DenseNet classification model concatenates the channels while maintaining all statistics leading to motivating feature reclaim. Hence, it has generated reasonable outputs related to the ResNet classifier. The performance of DenseNet is better than ResNet in terms of accuracy (87.5%), sensitivity (98.8%), precision (84.4%), and JSS (79%) due to the capacity of DenseNet in obtaining more paradigmatic and discriminative features for more accurate segmentation.

Table 2: The results obtained from various classifiers from datasets

Algorithm		KMC dataset						TCGA-LIHC dataset					
		ACC	SEN	SPE	PRE	JSS	ρ -value	ACC	SEN	SPE	PRE	JSS	ρ -value
ResNet	Mean	0.864	0.981	0.680	0.828	0.765	0.048	0.857	0.834	0.674	0.862	0.818	0.047
	SD	0.021	0.011	0.021	0.042	0.076	0.015	0.026	0.016	0.026	0.047	0.081	0.020
DenseNet	Mean	0.875	0.988	0.677	0.844	0.790	0.073	0.862	0.877	0.769	0.909	0.849	0.110
	SD	0.020	0.008	0.026	0.043	0.059	0.031	0.025	0.013	0.031	0.048	0.064	0.036
I-ResNetV2	Mean	0.887	0.988	0.690	0.862	0.852	0.064	0.882	0.942	0.789	0.896	0.849	0.130
	SD	0.035	0.010	0.041	0.038	0.063	0.015	0.040	0.015	0.046	0.043	0.068	0.020

(Continued)

Table 2 (continued)

Algorithm	KMC dataset						TCGA-LIHC dataset						
	ACC	SEN	SPE	PRE	JSS	ρ -value	ACC	SEN	SPE	PRE	JSS	ρ -value	
IRRCN	Mean	0.895	0.988	0.722	0.868	0.842	0.044	0.868	0.936	0.793	0.880	0.830	0.045
	SD	0.048	0.010	0.033	0.045	0.019	0.021	0.053	0.015	0.038	0.050	0.024	0.026
BreastNet	Mean	0.903	0.988	0.739	0.879	0.870	0.045	0.901	0.958	0.768	0.905	0.871	0.049
	SD	0.024	0.007	0.031	0.029	0.018	0.014	0.029	0.012	0.036	0.034	0.023	0.019
MicroNet	Mean	0.918	0.994	0.770	0.894	0.889	0.043	0.904	0.968	0.759	0.901	0.876	0.048
	SD	0.018	0.005	0.029	0.038	0.030	0.017	0.034	0.017	0.041	0.039	0.028	0.024
CBAM	Mean	0.926	0.994	0.791	0.904	0.899	0.043	0.912	0.964	0.788	0.916	0.885	0.049
	SD	0.018	0.004	0.034	0.028	0.024	0.016	0.023	0.009	0.039	0.033	0.029	0.021
ELM	Mean	0.891	0.994	0.686	0.912	0.859	0.044	0.890	0.952	0.747	0.896	0.857	0.042
	SD	0.019	0.003	0.023	0.028	0.034	0.018	0.028	0.014	0.044	0.038	0.034	0.026
LiverNet	Mean	0.930	0.994	0.800	0.910	0.905	0.039	0.915	0.964	0.797	0.921	0.890	0.032
	SD	0.019	0.005	0.021	0.025	0.019	0.020	0.033	0.019	0.049	0.043	0.039	0.031
ECNet	Mean	0.965	0.994	0.897	0.957	0.952	0.019	0.963	0.975	0.932	0.975	0.951	0.016
	SD	0.011	0.003	0.026	0.028	0.009	0.008	0.016	0.008	0.031	0.033	0.014	0.013

I-ResNet-v2 classifier realized better classification performance related to the other classifiers like ResNet and DenseNet. I-ResNet-v2 model substituted the kernel concatenation of inception with residual skip-connections to take the advantages of the ResNet and DenseNet models while maintaining the same level of processing overhead. It achieves 88.7% accuracy, 98.8% sensitivity, 69% specificity, 86.2% precision and 85.2% JSS. The dropout technique used in the ELM model can improve the prediction stability of the classification process. It provides a similar classification performance related to I-ResNetV2. The IRRCN model exploits the recurrent convolutional block with the inception and the residual network blocks. This model yields improved performance with an accuracy of 89.5%, sensitivity of 98.8%, specificity of 72.2%, precision of 86.8%, and JSS of 84.2%. The BreastNet exploits the bilinear upsampling layer, which enables the classifier to obtain attributes from various depths. Hence, this model achieves better results in terms of accuracy, sensitivity, specificity, precision, and JSS with 90.3%, 98.8%, 73.9%, 87.9%, and 87%, respectively. The MicroNet exploits skip-connections to assimilate lower-level attributes and efficiently combine them with multi-level attributes through a biased loss function. In terms of accuracy, sensitivity, specificity, precision, and JSS, MicroNet showed 91.8%, 99.4%, 77%, 89.4%, and 88.9%, respectively. The CBAM classifier achieves much higher classification performance with an accuracy of 92.6%, sensitivity of 99.4%, specificity of 79.1%, precision of 90.4%, and JSS of 89.9%.

The classification model using LiverNet hit better results with 93% accuracy, 99.4% sensitivity, 80% specificity, 90% precision, and 90.5% JSS. The classification performance of the proposed ensemble classifier, ECNet is usually better than the classification performance of each base learner used in ensemble systems such as DrpXLM and CBAM. As an average of 10-fold trials, the ECNet classifier achieved better results with an accuracy of 96.5%, sensitivity of 99.4%, specificity of 89.7%, precision of

95.7%, and JSS of 95.2%. The classification accuracy of the ECNet model is significantly greater by 8.35% than the individual DrpXLM classifier and slightly greater by 3.8% than the CBAM-based classifier as shown in Figs. 3 and 4. It can be witnessed that the SD of the ECNet was less than all other classifiers. Hence, the ECNet classification model delivers much more reliable outcomes for identifying liver cancer than the others. More precisely, ECNet is a very viable method for detecting liver cancer.

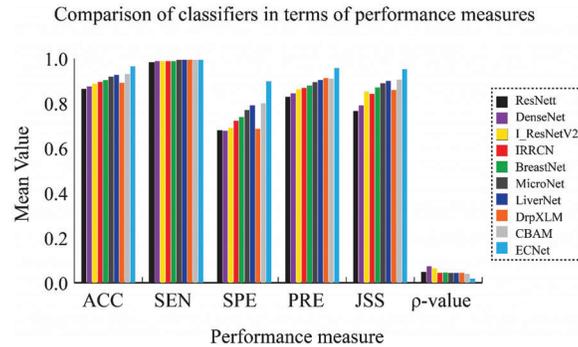


Figure 3: Comparison of results obtained from the KMC dataset in terms of the mean value of performance measures

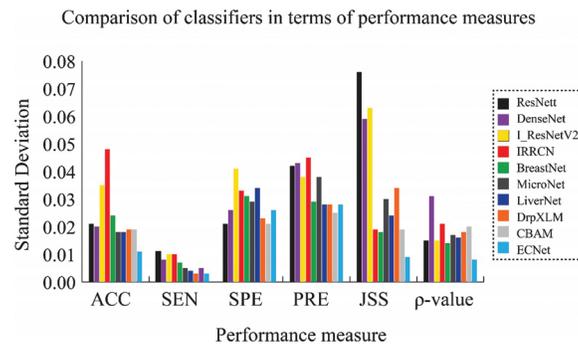


Figure 4: Comparison of results obtained from the KMC dataset in terms of SD of performance measures

We can achieve related results when we implement our model on the TCGA-LIHC liver dataset. The mean value of evaluation metrics and SD obtained from the TCGA-LIHC liver cancer dataset are given in Figs. 5 and 6. It is possible to conclude that the ECNet model has realized much-improved performance with 96.3% accuracy, 97.5% sensitivity, 93.2% specificity, 97.5% precision, and 95.1% JSS. Similarly, it is also witnessed that the SD obtained by the ECNet is lesser than that of all most all other classifiers which signify that the ECNet can yield more consistent and credible diagnosis results. Figs. 3–6 also display the Wilcoxon rank-sum test results. In most cases, it can be easily observed that the ρ values measured by the rank-sum test are less than 5% which demonstrates the superior enactment of ECNet. The proposed ECNet classifier decreases the time consumption of training and testing procedures. It consumes only 12.22 and 1.24 s, respectively in the KMC dataset. In the case of the TCGA-LIHC dataset, it takes 14.28 and 1.56 s for the training and testing process respectively. From this extensive empirical analysis, we can conclude that our ECNet is the better liver cancer diagnostic model related to state-of-the-art classifiers. However, this model does not consider the false alarm rate of the classification.

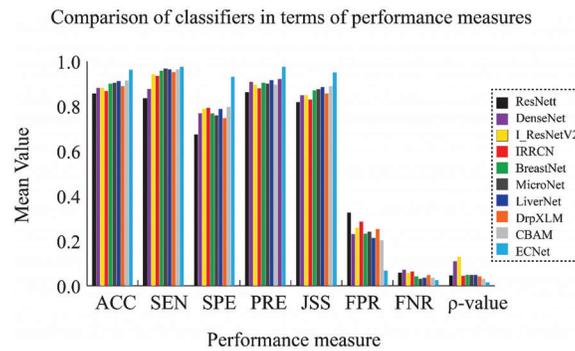


Figure 5: Comparison of results obtained from the TCGA-LIHC dataset in terms of the mean value

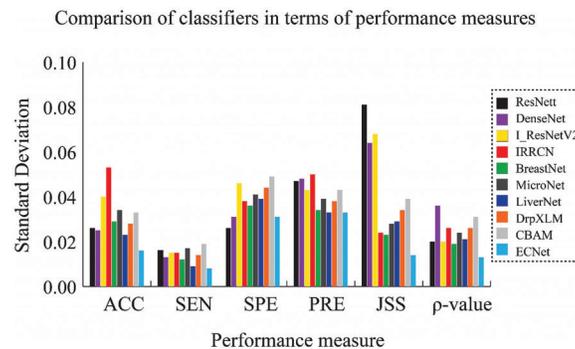


Figure 6: Comparison of results obtained from the TCGA-LIHC dataset in terms of the SD value

7 Conclusions

Through this work, we develop a very efficient deep ensemble classifier to categorize H&E liver histopathology images. ECNet applies a voting mechanism to integrate the decisions of individual classifiers using the average of probabilities rule. Initially, the nuclei regions in the image are segmented through SrCN, and then the segmented nuclei regions are fed into the ensemble DL network for classification. The empirical results obtained from the proposed model are related to some state-of-the-art classifiers in terms of performance metrics. The results of our meticulous experiments on the KMC liver dataset reveal that the proposed ECNet significantly outperforms other prevailing classification networks with better accuracy, sensitivity, specificity, precision, and JSS. Also, we carry out the Wilcoxon rank-sum test to determine whether the ECNet provides a considerable performance improvement or not. From extensive empirical analysis, we can conclude that our ECNet is the better liver cancer detection model related to state-of-the-art classifiers. We believe that this work will create cognition and interest in the field of liver cancer detection and related disease management. However, this work does not focus on false alarm rate and timing complexity of the intended algorithm. Moreover, the classification accuracy depends on the dataset used for experimentation. Hence, we plan to develop more generalized detection model with superior classification performance. Also, we plan to analyze the false alarm rate and timing complexity related to this proposed model in detail.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] I. H. Sarker, “Machine learning: Algorithms, real-world applications and research directions,” *Springer Nature Computer Science*, vol. 2, pp. 160–168, 2021.
- [2] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram *et al.*, “Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries,” *A Cancer Journal for Clinicians*, vol. 71, no. 3, pp. 209–249, 2021.
- [3] J. M. Llovet, R. K. Kelley, A. Villanueva, A. G. Singal, E. Pikarsky *et al.*, “Hepatocellular carcinoma,” *Nature Reviews Disease Primers*, vol. 7, no. 6, pp. 321–330, 2021.
- [4] S. Diwakar, A. N. Srinivas and P. K. Divya, “Etiology of hepatocellular carcinoma: Special focus on fatty liver disease,” *Frontiers in Oncology*, vol. 10, pp. 601710, 2020.
- [5] A. A. Aatresh, K. Alabhya, S. Lal, J. Kini and P. U. Saxena, “LiverNet: Efficient and robust deep learning model for automatic diagnosis of sub-types of liver hepatocellular carcinoma cancer from H&E stained liver histopathology images,” *International Journal of Computer Assisted Radiology and Surgery*, vol. 16, no. 9, pp. 1549–1563, 2021.
- [6] C. Yao, S. Jin, M. Liu and X. Ban, “Dense residual transformer for image denoising,” *Electronics*, vol. 11, pp. 418, 2022.
- [7] S. M. Răboacă, C. Dumitrescu, C. Filote and I. Manta, “A new adaptive spatial filtering method in the wavelet domain for medical images,” *Applied Sciences*, vol. 10, no. 16, pp. 5693, 2020.
- [8] M. Z. Hoque, A. Keskinarkaus, P. Nyberg and T. Seppänen, “Retinex model based stain normalization technique for whole slide image analysis,” *Computerized Medical Imaging and Graphics*, vol. 90, pp. 101901, 2021.
- [9] S. J. Zhou and X. Li, “Feature engineering vs. deep learning for paper section identification: Toward applications in Chinese medical literature,” *Information Processing and Management*, vol. 57, no. 16, pp. 313–324, 2020.
- [10] D. X. Gu, K. X. Su and H. M. Zhao, “A case-based ensemble learning system for explainable breast cancer recurrence prediction,” *Artificial Intelligence in Medicine*, vol. 107, pp. 101858, 2020.
- [11] E. Jussupow, K. Spohrer, A. Heinzl and J. Gawlitza, “Augmenting medical diagnosis decisions? An investigation into physicians’ decision-making process with artificial intelligence,” *Information Systems Research*, vol. 32, no. 3, pp. 713–715, 2021.
- [12] Y. Chai, Y. Bian, H. Liu, J. Li and J. Xu, “Glaucoma diagnosis in the Chinese context: An uncertainty information-centric Bayesian deep learning model,” *Information Processing and Management*, vol. 58, pp. 102454, 2021.
- [13] C. J. Tseng, C. J. Lu, C. C. Chang, G. D. Chen and C. Cheewakriangkrai, “Integration of data mining classification techniques and ensemble learning to identify risk factors and diagnose ovarian cancer recurrence,” *Artificial Intelligence in Medicine*, vol. 78, no. 1, pp. 47–54, 2017.
- [14] Y. Cao, T. A. Geddes, J. Y. H. Yang and P. Y. Yang, “Ensemble deep learning in bioinformatics,” *Nature Machine Intelligence*, vol. 2, no. 9, pp. 500–508, 2020.
- [15] L. Dongdong, C. Ziqiu, W. Bolu, W. Zhe, Y. Hai *et al.*, “Entropy-based hybrid sampling ensemble learning for imbalanced data,” *International Journal of Intelligent Systems*, vol. 36, no. 7, pp. 3039–3067, 2021.
- [16] Y. A. LeCun, L. Bottou, G. B. Orr and K. -R. Muller, “Efficient backprop” in *Neural Networks Tricks of the Trade*, 2nd ed., vol. 7700, Berlin, Heidelberg: Springer, pp. 9–48, 2012.
- [17] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in *Proc. 27th Int. Conf. on Machine Learning*, Haifa Israel, HI, Israel, pp. 807–814, 2010.
- [18] G. Huang, Z. Liu, L. V. D. Maaten and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc. 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 4700–4708, 2017.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed *et al.*, “Going deeper with convolutions,” in *Proc. 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, CVPR, Boston, MA, pp. 1–9, 2015.
- [20] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proc. 32th Int. Conf. on Machine Learning*, Lille, France, pp. 448–456, 2015.

- [21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 2818–2826, 2016.
- [22] C. Szegedy, S. Ioffe, V. Vanhoucke and A. A. Alemi, "Inception-V4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31 AAAI Conf. on Artificial Intelligence*, San Francisco, California, USA, pp. 4278–4284, 2017.
- [23] C. Ferreira, T. Melo and P. Sousa, "Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2," *Image Analysis and Recognition*, vol. 10882, pp. 763–770, 2016.
- [24] M. Z. Alom, C. Yakopcic, T. M. Taha and V. K. Asari, "Breast cancer classification from histopathological images with inception recurrent residual convolutional neural network," *Journal of Digital Imaging*, vol. 32, no. 4, pp. 605–617, 2019.
- [25] M. Togaçar, K. B. Özkurt, B. Ergen and Z. Cömert, "Breastnet: A novel convolutional neural network model through histopathological images for the diagnosis of breast cancer," *Physica A: Statistical Mechanics and Its Applications*, vol. 545, pp. 123592, 2020.
- [26] S. E. A. Raza, L. Cheung, M. Shaban, S. Graham, D. Epstein *et al.*, "Micro-Net: A unified model for segmentation of various objects in microscopy images," *Medical Image Analysis*, vol. 52, no. 1, pp. 160–173, 2019.
- [27] S. Woo, J. Park, J. -Y. Lee and I. S. Kweon, "CBAM: Convolutional block attention module," in *Lecture Notes in Computer Science*, vol. 11211, Springer, Cham, pp. 3–19, 2018.
- [28] A. Dogantekin, F. Özyurt, E. Avcı and M. Koç, "A novel approach for liver image classification: PH-C-ELM," *Measurement*, vol. 137, no. 7, pp. 332–338, 2019.
- [29] A. A. Aatresh, K. Alabhya, S. Lal, J. Kini and P. U. Saxena, "LiverNet: Efficient and robust deep learning model for automatic diagnosis of sub-types of liver hepatocellular carcinoma cancer from H&E stained liver histopathology images," *International Journal of Computer Assisted Radiology and Surgery*, vol. 16, no. 9, pp. 1549–1563, 2021.
- [30] J. Derrac, S. García, D. Molina and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.