Received: July 11, 2022. Revised: August 21, 2022. 

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

# CATDSNet: Computer Aided Tongue Diagnosis System for Disease Prediction Using Hybrid Extreme Learning Machine

**Sreerama Prasad Chelluboina[1]*** **Kunjum Nageswara Rao[1]**

[1]*Department of Computer Science and Systems Engineering, Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India*
* Corresponding author's Email: srpchelluboina@gmail.com

**Abstract:** Tongue diagnosis is an important way of diagnosing the human health in Indian ayurvedic medicine (IAM), which helps to identify different diseases through tongue image analysis. Traditionally, this diagnosis system has been done by ayurvedic experts to recognize the type of health issue by observing the human tongue. But the recent advancements in machine learning algorithms in medical field enabled the researchers to implement a computer aided diagnosis system for ayurvedic medical treatment. However, the existing tongue diagnosis systems are suffering with poor classification performance due to the variations in tongue appearance such as color, shape, coating, and texture properties. Therefore, this article focuses on computer aided tongue diagnosis system (CATDS) for disease predication through tongue image analysis, here after denoted as CATDSNet. Initially, fast nonlocal mean (FNLM) filtering is applied on test image to perform the preprocessing of given tongue dataset. Next, color features are extracted from the pre-processed tongue images using colour moments. In addition, grey level cooccurrence matrix (GLCM) is used to extract the texture features information. Then, this extracted color, and texture features are used to train the proposed CATDSNet using hybrid extreme learning machine (HELM) classifier for classifying various diseases such as healthy, appendicitis, bronchitis, gastritis, heart disease, and pancreatitis. The obtained simulation results on tongue image dataset disclose the superiority of proposed CATDSNet model as compared to state-of-the art approaches such as random forest, support vector machine (SVM), and SVM-based recursive feature elimination (SVM-RFE) with a classification accuracy of 92.3%, precision of 92.39%, recall of 92.06% and F1-score of 92.02%.

**Keywords:** Indian ayurvedic medical treatment, Tongue image analysis, Grey level cooccurrence matrix, Nonlocal means, Extreme learning machine.

## 1. Introduction

In oriental medicine, such as IAM, traditional Chinese medicine, Japanese traditional herbal medicine, and traditional Korean medicine, the condition of a patient's internal organ can be effectively evaluated using a method that does not involve any sort of invasive procedure through the use of tongue diagnosis [1]. The procedure of diagnosis is dependent on the professional's opinion, which is derived from a visual examination that includes the colour, material, coating, shape, and motion of the tongue. Traditional tongue diagnosis is more likely to detect the condition than it is to notice abnormalities in the look of the tongue and diseases that affect the tongue. For instance, a white, greasy coating on the tongue might indicate cold syndrome, while a yellow, thick coating on the tongue can indicate hot syndrome [2]. Both of these syndromes are linked to health issues such as infection, inflammation, stress, immunological diseases, or endocrine abnormalities. According to IAM while, the distinct parts of the tongue are shown in Fig. 1 along with the internal organs that correlate to those parts of the tongue. Pathological changes in the heart and lungs are reflected in the tip of the tongue, whereas abnormalities in the liver and gallbladders are reflected on the sides of the tongue on both sides of the tongue. The center of the tongue reflects the pathological changes that have occurred in the spleen

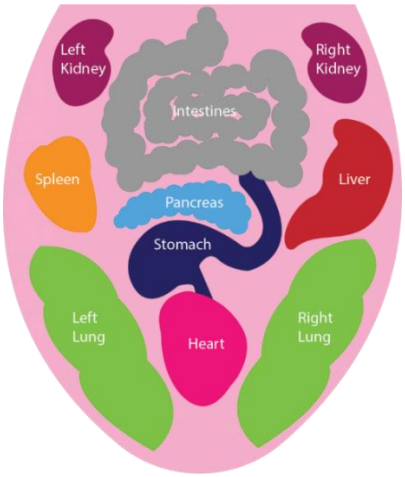



Figure. 1 Reflex zones of Tongue

and stomach, while the root of the tongue reflects the pathological abnormalities that have occurred in the kidneys, intestines, and bladder region [3].

Traditional tongue diagnosis, on the other hand, is heavily dependent on the amount of clinical expertise a clinician has; as a consequence, various doctors are likely to arrive at different diagnostic conclusions for the same patient [4]. Thankfully, computer-aided methods for tongue diagnosis are able to improve upon these shortcomings because to the utilization of computers and other approaches that are relevant to the field [5]. The objective, quantitative, and automated auxiliary analyses that have been introduced to contemporary tongue diagnosis as a result of developments in computer science and technology have assisted medical professionals in making more accurate diseases diagnoses. The modernization of the IAM tongue diagnostic has mostly resulted in the technique being broken down into three parts: segmentation of the tongue, extraction of tongue features, and analysis of the condition. In order to get an accurate image of the tongue from the initial image of the tongue, segmentation of the tongue makes use of a segmentation algorithm [6]. Because digitally acquired tongue images include not only the tongue region but also parts of the none-tongue regions such as lips, teeth, and so on, tongue segmentation from the whole image is a prerequisite for tongue characterization. Digitally acquired tongue images include both the tongue region and parts of the none-tongue regions. Nevertheless, tongue segmentation is a challenging process because of interference caused by shifting light and crowded backgrounds. After that, the characteristics of the tongue that are necessary for an IAM diagnosis are extracted. After the traits have been retrieved, they are utilized to categories and diagnose diseases. In recent years, approaches for tongue image feature extraction have been subjected to considerable research [7]. According to the findings of these investigations, the approaches of computer-assisted tongue diagnosis may be broken down into two distinct categories: single feature and multifeatured.

There have been many different proposals for and applications of single feature extraction approaches for tongue image analysis [8]. These kinds of algorithms may extract important information based on a very basic descriptor, such as the color, texture, shape, or orientation of an object. The identification of the sort of tongue image that may assist IAM clinicians in reaching a further diagnostic conclusion is the work that is referred to as the tongue image-based diseases categorization. The segmentation of the tongue has been addressed from a variety of angles and approaches in the past [9]. Some of these methods are sensitive to changes in illumination or backgrounds that are clustered, some of these methods confuse the lips with the body of the tongue, and some of these methods require additional preprocessing, which makes the entire segmentation process more complicated. However, the majority of these methods are based on traditional image processing techniques [10]. In more recent times, automated tongue segmentation has been approached from the perspective of deep learning-based approaches.

Although the performance of those approaches based on deep learning is superior to the performance of the most conventional methods for segmenting the tongue, those methods still have significant limitations. Image enhancement is an example of the additional preprocessing that is necessary, which contributes to the overall complexity of the segmentation process. In a similar way, brightness discrimination as a preprocessing step decreases a model's capacity for generalization based on deep learning [11]. Nevertheless, mask is able to locate the item; however, it is unable to differentiate between the different types of objects. As a consequence, the segmentation process is sluggish and less accurate than it might be since unrelated objects were processed unnecessarily.

The following is a list of the primary goals of this work to achieve in order to address these issues:

- Implementation of CATDSNet model to classify the healthy, appendicitis, bronchitis, gastritis, heart disease, and pancreatitis diseases using HELM classifier.
- Initially, FNLM is applied to remove the noises from the tongue images, which also performs the color enhancements.

- Implementation of joint texture and color features are extracted from pre-processed image using color moments and GLCM descriptor.
- Finally, HELM is used to classify the diseases using extracted color, and texture features.

Rest of the paper is contributed as follows: section 2 deals with the related work with their drawbacks. Section 3 describes the detailed analysis of proposed CATDS model. Section 4 explains the simulation results and discussion with comparative analysis. Section 5 deals with the conclusion and future scope followed by references.

## 2. Related work

This section is focused on survey of tongue disease detection and classification methods. Initially, machine learning models such as logical regression, random forest [12], support vector machine (SVM) and naive bayes are widely used to classify the early-stage oral tongue squamous cell cancer. Here, SVM method resulted in better accuracy as compared to other machine learning models [13]. Further, deep learning-based feedforward neural network (FFNN) is also used to classify the OTSCC but resulted in the reduced accuracy such as 92.7% [14]. In [15], authors developed the deep convolutional neural network (DCNN) model for classifying the benign and pre-cancerous diseases from tongue image. But this method suffering with the high computational complexity due to high training time. Further, DCNN also used for tongue health classification with ResNet34 feature extraction [16]. Anyhow, this method is capable of classifying healthy and unhealthy nature of tongue, but unable to classify the diseases from tongue images. Recently, tongue diagnosis analysis system (TDAS) is proposed for detection of prediabetes and diabetes diseases from tongue images [17]. The TDAS system contains ResNet-50 for extracting the colour and textures features with SVM classification. However, this method is suffering with the high computational complexity. Further, deep learning based radial basis function neural network (RBFNN) based classifier is used for diabetes disease classification from tongue images [18]. Here, ResNet50 us used to extract the colour, shape, thickness, and tooth marked based features. Anyhow, this method suffers with the low classification accuracy with higher training time.

In [19] authors proposed automated synergic deep learning-based tongue color image (ASDL-TCI) for tongue disease classification. Here, median filtering is used to perform the preprocessing operation with ASDL feature extraction. Finally, deep neural network (DNN) based classifier is used with enhanced black widow optimization (EBWO) optimization. But this method is unable to classify the multiple diseases from tongue images. In [20] authors used the machine learning models for classify the diabetes mellitus and symptoms of gastric disease from tongue images. Here, SVM recursive feature elimination (SVM-RFE) is used to extract the optimal color and texture features. Further, fully-channel regional attention network is developed for the disease identification and classification using multi-layer convolution networks [21]. However, this method extracted the basic tongue features. To overcome these problems, hybrid deep learning models are developed by combining the individual CNN models with ensemble stacking [22]. Anyhow, this method identifies the diseases using physique analysis of tongue, which resulted in poor performance. Further, Zero-Shot learning for constitution recognition (ZSLCR) is implemented with wResNet18 and ResNet34 models for tongue color analysis [23]. This method utilizes the automatic weight updation properties, which resulted in the better performance but taking much time for training. In [24] authors implemented the diabetic risk prediction using hybrid machine learning classifier through tongue images. This work considered the ResNet50 for extracting the features from tongue images. However, there are huge loss of features during the training procedure, which resulted in reduced performance. In addition, the hybrid features are extracted using transfer learning models, which analyzed the patterns of the tongue images and extracted hided features [25]. Further, color-based disease analysis is performed using this method, which is not considered the feature selection operations.

However, various works are carried out on single, two diseases diagnosis (classification) from tongue images, but multi class disease classification from tongue images is still a challenging research problem. Further, as per authors knowledge, there is no work has been published on multi class disease classification from tongue images.

## 3. Proposed methodology

Disease prediction from tongue images is difficult task as the tongue images contains different colours, structures. Therefore, this work is focused on implementation of hybrid machine learning models for disease prediction through tongue images. Fig. 2 presents the proposed CATDSNet architecture for ayurvedic medical treatment. Initially, FNLM

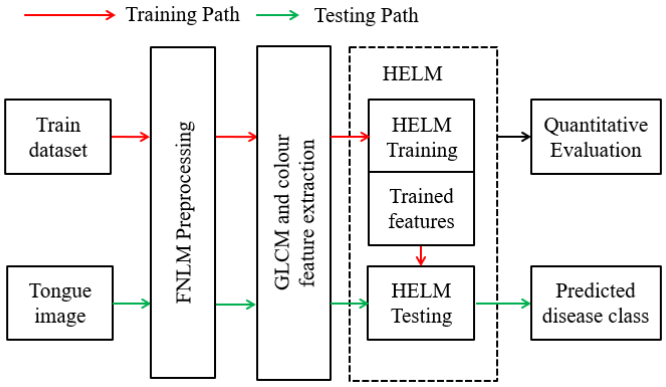



Figure. 2 Block diagram of proposed CATDSNet model

preprocessing method is performed for the noise removal and overcoming the colour illumination problems. Next, color features such as mean, skewness, and standard deviation are extracted from these pre-processed tongue images using colour moments. Additionally, GLCM is used to extract the texture features like energy, contrast, entropy, correlation, and homogeneity etc. from pre-processed tongue images. Finally, HELM classifier is used to classify various diseases healthy, appendicitis, bronchitis, gastritis, heart disease, and pancreatitis disease classes from extracted color, and texture features.

### 3.1 Pre-processing

Usually, the tongue images are captured in the different conditions, which contains the dissimilar properties like shape, size, rotation angle, resolution, number of pixels, foreground light, background intensities, colour, contrast, brightens, hue-based colour properties. The uneven nature of these properties makes the feature extraction more difficult, which causes reduction of performance. Further, the people may contain tongue infections, tongue cancers, and some other tongue diseases, which resulted in mismatch in disease prediction operation. Thus, image preprocessing operation is performed to overcome these problems, which normalizes the properties of all images equally. Further, image preprocessing also removes the basic background and foreground illumination problems of the tongue images. Usually, tongue images are damaged by several forms of artifacts, which lowers the overall performance of CATDSNet. Here in this work, FNLM is successfully used to remove the distortions from tongue images. Further, FNLM also increases the contrast, brightness, and color based statical qualities of tongue images. For instance, consider the $f(n,m)$ is the noisy tongue image, which is created by adding the noise to original image. The noise model of given tongue image is formulated as follows.

$$f(n,m) = v(n,m) * u(n,m) + \gamma(n,m) \quad (1)$$

In this case, $n$ and $m$ stands the rows and columns of the tongue image, which may also be thought of as spatial coordinates.

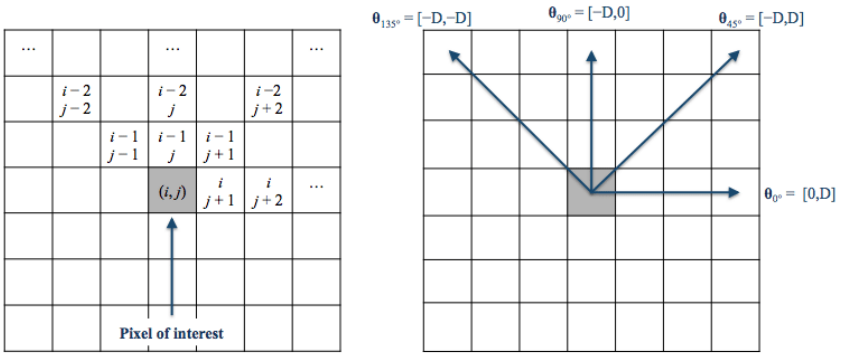
In addition, $u(n,m)$ is the representation of the original image without any noise, $\gamma(n,m)$ is the representation of the artefacts that include a variety of noises, and $v(n,m)$ is the representation of the damaged image pixel characteristics. The FNLM operation is carried out on $f(n,m)$ by using the non-local-mean window, which is denoted by $w(n,m)$. Here, FNLM carries out the convolution operation between $f(n,m)$ and the overlapping bands of $w(n,m)$, which ultimately results in the generation of a noise-free output image denoted by u $o(n,m)$, respectively. The process of FNLM is depicted n following manner.

$$u_o(k,l) = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} w(k-n,l-m) * f(n,m) \quad (2)$$

The maximum allowed rows in this instance are N, while the maximum allowed columns are M. In addition, the most important part of the noise reduction process is the construction of the FNLM window $w(n,m)$, which uses linear exponential probability models to generate weight coefficients. This is where the majority of the work is done.

$$w(k-n,l-m) = \frac{1}{Z[n,m]} e^{-\frac{\left\|N_i[P_{n,m}]-N_j[P_{k,l}]\right\|_{2a}^2}{h}} \quad (3)$$

Here, $N_i$ , $N_j$ are the pixel patches that are located in the immediate neighborhood of the window. Further, $a$ represents the non-local gaussian kernel-based standard deviation. $Z[n,m]$ represents the zero mean of the window values. h represents the filtering




(a) (b)

Figure. 3 GLCM feature extraction: (a) spatial relationships of pixels and (b) GLCM directions

parameter of the window. In addition, the weight ranges are chosen from those that fall within the stated range, such as 0 to $w(N_i, N_j)$ less than 1. In addition, the non-local kernel pixels keep the uniform property, such as $P_j \in N_i\ w(N_i, N_j) = 1$; this is the case because of the non-local nature of the kernel. In addition, $Z[n,m]$ may be obtained using the following formula:

$$Z[n,m] = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} e^{-\frac{\|N[P_n,m]-N[P_k,l]\|^2{}_{2,a}}{h}} \quad (4)$$

In order to determine which pixel patches are comparable to one another, a calculation called the Euclidean distance between $N_i$ , $N_j$ patches are performed.

$$d = \left\|I(N_i) - I(N_j)\right\|^2_{2,a} \quad (5)$$

Then, the weight coefficients are revised using the aforementioned weight coefficients in the following manner:

$$w(N_i, N_j) = \frac{1}{Z(i)} e^{(-d)/h^2} \quad (6)$$

Finally, the denoising and enhancement procedure is completed by taking into account this updated $w(N_i, N_j)$.

$$u_o(k,l) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} w(N_i, N_j) \times f(n,m) \quad (7)$$

### 3.2 Feature extraction

Features are the statistical properties of any image, which holds the colour, shape, entropy, and other forbidden characteristics of image. Further, the appropriate feature extraction will result in improvement of the classification performance. The proposed CATDS model extracts the GLCM based texture features, and the color features such as mean, skewness, standard deviation.

#### 3.2.1 GLCM feature extraction

Techniques for analysing images include the general linear correspondence method and related texture feature computations. The GLCM is a tabulation that determines how frequently various combinations of grey levels co-occur in an image or image sub region as shown in Fig. 3. Given an image that is composed of pixels, each of which has an intensity (a particular grey level). Then, the GLCM is applied to determine how often different combinations of grey levels co-occur. Calculations using texture features make use of the information contained within the GLCM to provide a measure of the change in image texture (also known as intensity) at the pixel of interest. Quantize the image data, so the value of each sample on the echogram is considered to be the intensity of the corresponding image pixel, and each sample is regarded as a single pixel on the echogram. After then, the intensities are further quantized into a certain number of discrete grey levels.

The GLCM dimensions will be $N$ squares on each side, where $N$ is the number of levels that was given under the quantization heading. Let us assume that $S$ is the sample that will be used for the computation. Let $W$ represent the collection of samples that surround sample s and that fit inside a window that is centered on sample s and has the size that was defined under the heading window size. Taking into account just the samples that are part of the set $W$ with elements $i, j$, where these two samples with intensities $i\ and\ j$ occur in a certain spatial relationship (where $i$ and $j$ are intensities between 0 and $N-1$). The total number of instances in which the required spatial connection may be found in $W$ will be equal to the sum of all of the components $i, j$

that make up the GLCM. Create a symmetrical GLCM and continue the operation for all pixels.

Create a duplicate of the GLCM that has been transposed with different phase angle. The GLCM will compute the surrounding pixels in $0^0$, $45^0$,$90^0$,$135^0$, and $180^0$ angles. This results in a symmetric matrix in which the connection $i$ to $j$ cannot be differentiated from the relationship $j$ to $i$ due to the symmetry of the matrix (for any two intensities $i$ and $j$). The components of the GLCM may now be interpreted as probabilities of discovering the connection $i,j$ (or $j,i$ in $W$. The value of this computed feature is substituted for the sample $S$ in the virtual variable that was produced as a consequence. The texture features extracted using GLCM are energy, entropy, contrast, homogeneity, and correlation.

$$Energy = \sum_{i,j=0}^{N-1}\left(P_{ij}\right)^2 \quad (8)$$

$$Entropy = \sum_{i,j=0}^{N-1} -\ln\left(P_{ij}\right)P_{ij} \quad (9)$$

$$Contrast = \sum_{i,j=0}^{N-1}\left(P_{ij}(i-j)\right)^2 \quad (10)$$

$$Homogeneity = \sum_{i,j=0}^{N-1}\frac{P_{ij}}{1+(i-j)^2} \quad (11)$$

$$Correlation = \sum_{i,j=0}^{N-1} P_{ij} \times \frac{(i-\mu)\,(j-\mu)}{\sigma^2} \quad (12)$$

Here, $P_{ij}$ is the elements of the symmetrical GLCM features. Here, $\mu$ is GLCM mean, it is a thought of an estimate of the intensity of each pixel that participated in the associations. This also approximates, but is not equivalent to, the mean of all the pixels in the data window $W$, and it is reliant upon the choice of spatial connection such as direction. Further, $\sigma^2$ is the variation in intensity across all reference pixels used in the relationships that were used to calculate the GLCM.

$$\mu = \sum_{i,j=0}^{N-1} i \times P_{ij} \quad (13)$$

$$\sigma^2 = \sum_{i,j=0}^{N-1} P_{ij}(i-\mu)^2 \quad (14)$$

#### 3.2.2 Color features

The colour features hold the colour intensities of tongue image, which are extracted using colour moments. Moments of colour are measurements that define the colour distribution in an image in the same way as central moments uniquely describe a probability distribution. Color moments are similar to central moments in that they characterise the distribution of colours in an image. Color moments are most often used for the purpose of colour indexing, where they serve as features in image retrieval applications and are used to evaluate the degree to which two images are comparable on the basis of colour. In most cases, one image is compared to a database of digital images that have already had their characteristics calculated in order to discover and get an image that is similar to the one being searched for. Following each comparison of two diseases, a similarity score is generated; the lower this value is, the greater the likelihood that the two images have a high degree of resemblance.

Color moments are the three most prominent points in an image's color distribution that are used. The mean ($\mu_c$), standard deviation ($\sigma_c$), and skewness are their respective features. The value of a color that is considered to represent the image's pixels average is referred to as the mean. The square root of the variance of the distribution is the analysis for calculating the standard deviation. A measure of the degree to which the distribution is not symmetrical is referred to as the skewness of the distribution. The value of a color that is considered to represent the image's mean is referred to as the mean. At least three different values may be used to define a color. In an image, moments are computed for each of these channels individually. Therefore, an image is defined by 9 moments, with 3 moments corresponding to each of its three-color channels. After that, the three-color moments may be characterized as follows:

$$\mu_c = \sum_{i,j=0}^{N-1}\frac{1}{N} \times P_{ij} \quad (15)$$

$$\sigma_c = \sqrt{\frac{1}{N}\left(\sum_{i,j=0}^{N-1}(P_{ij}-\mu_c)^2\right)} \quad (16)$$

$$S_i = \sqrt{\frac{1}{N} * \left(\sum_{i,j=0}^{N-1}(P_{ij}-\mu_c)^3\right)} \quad (17)$$

### 3.3 HELM classification

The bootstrap multi-layer aggregation, which is often referred to as multi-layer perception (MLP), and it is an ensemble strategy that is used to increase the accuracy and stability of an algorithm by minimizing the amount of time it spends classifying data. MLP is the process that is used to create sub-datasets from the primary training dataset. In this particular step of the division process, sampling and the replacement of datasets are used. The data is then trained using a variety of classifiers, and the results are predicted based on an average of all the samples that were put through the testing process. Due to the

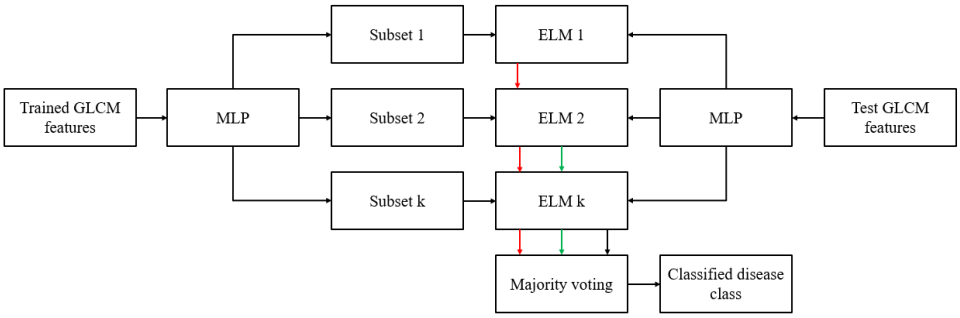



Figure. 4 Architecture of proposed HELM classifier

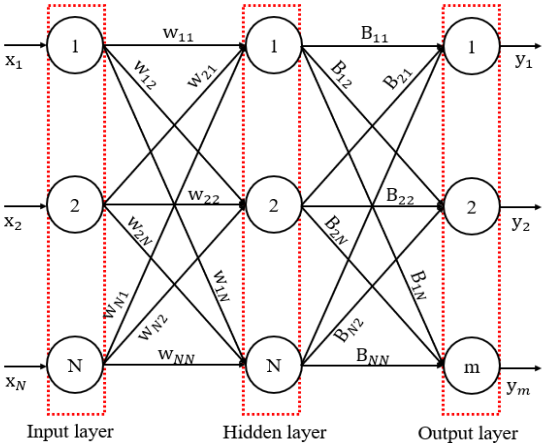



Figure. 5 MLP structure of HELM

ease with which it may be performed, the standard ELM is sometimes referred to as the "base learner" for the MLP approach. However, the ELM is unable to enhance the accuracy of week learners and is unable to detect week learners in sub training datasets. In addition, conventional ELM is unable to predict the best predicted class. This is because MLP helps to minimize the unpredictability of predictions, which makes it impossible for conventional ELM to predict the best predicted class. Therefore, the final decision in a classification model that makes use of MLP is arrived at via the employment of a voting method that requires a simple majority. The primary goal in integrating the MLP ensemble model into the HELM is to improve the accuracy and consistency of the categorization. Fig. 4 is an illustration of the HELM classifier.

The following is an illustration of how HELM performs its functions:

- The MLP module of HELM is responsible for partitioning the dataset into k groups consisting of a variety of different training data subsets.
- In order to gain $K$ trained HELM features as a consequence of the training process, first train $k$ ELM classifiers by using the proper training data subsets. These ELMs operate in a manner that is parallel to one another.
- During the training process, identify the weak learners, and then enhance the performance of the weak learners by modifying the dataset that is being applied.
- Calculate the projected disease class label for the test data set using the $k$ trained ELM classifier. Based on the test characteristics, ELM will provide $k$ prediction results.
- The last step in the ELM process is to perform a majority voting operation on each of the individual ELM classification results. This produces the definitive disease type classification.

Fig. 5 is a representation of the MLP structure of HELM. This structure has m number of output nodes, $N$ number of input nodes, and $N$ number of hidden nodes, and it is subdivided into three layers. Think of the input features that were extracted from the GLCM as $x = (x_1, x_2, \dots, x_N)^T$, where $N$ is the number of features, and it is going to be used on the input layer.

A collection of weights denoted by the notation $w_i$=$[w_{i1}, w_{i2}, \dots, w_{iN}]^T$ is the connection that is made between each of the nodes in the input layer and the nodes in the hidden layers, where $N$ is the total number of weights. Further, bias weights $B_j$ = $(B_{j1}, B_{j2}, \dots, B_{jN})^T$ are used in the process of interconnecting nodes of the output layer with nodes of the hidden layer.

The process is carried out by the hidden layer using the hidden matrix $H$, and it is described in the following way:

$$H = g\left(w_i^T x + b_i\right) \tag{18}$$

In this context, the activation function of HELM is denoted by $g(.)$, and the bias function of HELM is denoted by $b_i$. Then, the convolution operation is carried out between $B_j$ and $H$, which results in the generation of the anticipated vector

Table 1. HELM classification algorithm

| | |
|---|---|
| **Input:** | Trained features $s$, Test feature $\psi_{test}$ |
| **Output:** | Predicted disease class |
| | **Step 1:** Apply the GLCM-trained feature (s) to HELM's MLP classifier. |
| | **Step 2:** The MLP process will be started, and the dataset will be divided into many sub datasets. |
| | **Step 3:** Classify each individual sub dataset using the HELM algorithm. |
| **Training Phase** | |
| | **Step 4:** After going through the training procedure, generate the best possible value for the $B_j$ and $w_i$ weights. |
| | **Step 5:** Identify the week learners of the HELM classifier, then optimize those week learners using the most recent dataset. |
| | **Step 6:** The last step is to complete the hidden matrix for the output layer and the hidden layer by utilizing appropriate weights. |
| **Testing phase:** | |
| | **Step 7:** The HELM with MLP classifier may be improved by using the test features ($\psi_{test}$). |
| | **Step 8:** Apply the optimized $B_j$ and $w_i$ weights to testing model of HELM. |
| | **Step 9:** Improve the overall performance of each week's learners by adjusting the weights based on feedback. |
| | **Step 10:** a majority voting operation is carried out between the probabilities of the different sub dataset's classification performances for the final choice. |
| | **Step 11:** The classification of diseases classes is completed with the help of eq. (24). |

$\bar{y} = (\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_m)^T$. and the explanation behind this is as follows:

$$\bar{y} = \sum_{j=1}^{N} B_j \times H \tag{19}$$

The typical ELM uses a selection procedure in which the values for $B_j$ and $w_i$ are picked at random from a pool of potential values. These values are not created by a static training process. However, this led to a decrease in performance; as a consequence, this study changed a characteristic of ELM and developed HELM.

HELM produces the $B_j$ and $w_i$ weights from new training set $(s)$ by using the technique of reinforcement learning. In this case, the output and input vectors for the $k^{th}$ training instance are denoted as $y_k = [y_{k1}, y_{k2}, \ldots, y_{kk}]$ and $x_k = [x_{k1}, x_{k2}, \ldots, x_{kk}]$, respectively. In addition, the $B_j$ and $w_i$ weights are developed via the process of optimising the training set. The goal function of the optimization process is represented by Eq. (21), which has to be solved and minimized in order to provide an effective conclusion.

$$L(B, \zeta) = \frac{1}{2} \left\| B^2 \right\| + \frac{c}{2} \sum_{k=1}^{K} \left\| \zeta_k^2 \right\| \tag{20}$$

$$H(w_k) = L(yk - \zeta_k) \tag{21}$$

Here, $L()$ represents the feedback process, which is feedback to hidden layer from output layer, $c$ represents the regularization parameter, $\zeta_k$ represents the predicted error of instance $k$, $H(w_k)$ represents the hyperparameter of $w_i$.

$$H(w_k) = \begin{bmatrix} g(w_1^T x_k + b_1) \\ g(w_2^T x_k + b_2) \\ . \\ . \\ g(w_{\bar{N}}^T x_k + b_N) \end{bmatrix} \tag{22}$$

Applying the Kuhn–Tucker conditions, such as lagrange multipliers, serves the aim of resolving the optimization constraint that was mentioned before, and the solution that is obtained as a consequence is as follows:

$$\vartheta_{Bw} = \left( \frac{I_{N \times N}}{C} + \left( H(w_k) \right)^T H(w_k) \right)^{-1} \left( H(w_k) \right)^T \Upsilon \tag{23}$$

In this case, $\vartheta_{Bw}$ represents the value that has been optimised for the $\vartheta_{Bw}$weights, $I_{N \times N}$ stands for an identity matrix, and $\Upsilon$ is the output feedback constant. Finally, the layers of HELM were updated with optimized $\vartheta_{Bw}$ weights, and the output vector $y_k$ was produced.

$$y_k = \begin{cases} class\ A: & \rho^{class\ A} > \rho^{class\ B} \\ class\ B: & \text{else} \end{cases} \tag{24}$$

Here, $class\ A$ to $classB$ represents 6 classes including healthy, appendicitis, bronchitis, gastritis, heart disease, and pancreatitis diseases. Here,

Table 2. Dataset description.

| Class | Number of images |
|---|---|
| Healthy | 105 |
| Appendicitis | 126 |
| Bronchitis | 105 |
| Gastritis | 126 |
| Heart | 146 |
| Pancreatitis | 105 |
| Total | 713 |

Figure. 6 Sample tongue image dataset labelled with various diseases

$\rho^{class\ A}$ represents probability of $class\ A$ disease type and $\rho^{class\ B}$ represents probability of $class\ B$ disease type. Table 1 describes the overall stepwise algorithm of HELM classification model.

## 4. Results and discussion

This section gives the detailed analysis of simulation results and performance comparison with conventional approaches. Further, the proposed CATDS method and existing classifiers used the same dataset for evaluating the performance. The performance metrics are derived as follows:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (25)$$

$$Senisitivity = \frac{TP}{TP+FN} \quad (26)$$

$$specificity = \frac{TN}{TN+FP} \quad (27)$$

$$precision = \frac{TP}{TP+FP} \quad (28)$$

$$Recall = \frac{TP}{TP+FN} \quad (29)$$

$$F1 - Score = \frac{2.Precision.recall}{Precision + recall} \quad (30)$$

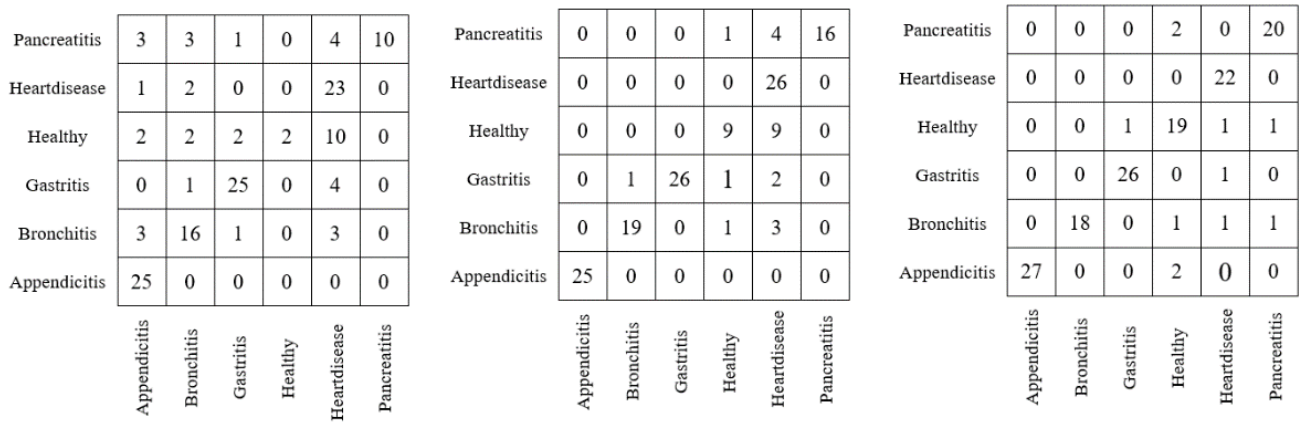
Here, $TP$ represents the true positive, which shows correctly predicting true labels (we anticipated "yes", and it's "yes"). Then, true negative (TN) represents the correctly predicting the other name (we anticipated "no", and it's "no"). Further, false positive (FP) represents the falsely predicting a label with true condition (we anticipated "yes", yet it's "no"). Moreover, false negative (FN) represents the falsely predicting a label with false condition (we anticipated "no", yet it's "yes")

### 4.1 Dataset

The dataset contains the healthy, appendicitis, bronchitis, gastritis, heart disease, and pancreatitis disease classes. As per authors knowledge, there is no standard dataset for disease prediction using tongue images. Therefore, the dataset used in this work is collected by various internet sources and followed the rules of IAM for disease indexing. Further, data augmentation also used to increase the number of samples during training with the usage of morphological operations, rotation, scaling, resizing, compression, flipping and masking, which can improve the performance of proposed CATDS method. Table 2 presents the number of images presented in the dataset w.r.t each disease class. Fig. 6 illustrate the sample tongue image dataset labelled with different diseases.

### 4.2 Performance evaluation

This section compares the performance evaluation of proposed CATDS model as compared to existing models. Fig. 7 presents the confusion matrices of SVM [13], random forest [12], and proposed CATDS using GLCM-based HLEM classifier, whereas Fig. 7 (a) presents the obtained confusion matrix using SVM-based CATDS while Fig. 7 (b), and Fig. 7 (c) discloses the obtained confusion matrices of CATDS using random forest, and proposed GLCM-based HELM classifier, respectively. It is noticed that existing SVM, and random forest classification models resulting in smaller number of true positives, and increased in false negative, whereas the proposed GLCM-based HELM classifier resulted in higher true positive values and less false predictions which in results the enhanced classification performance of proposed CATDS. Table 3 presents the performance comparison of proposed GLCM-based HELM classifier with existing SVM [13], random forest [12], TDAS [17], and SVM-RFE [20], where the proposed CATDS model outperforms existing CATDS models in terms of quality metrics such as accuracy, precision, recall, $F_1$-score, sensitivity, and specificity.




(a) (b) (c)

Figure. 7 Obtained confusion matrix using: (a) SVM [13], (b) random forest [12], and (c) proposed GLCM-based HLEM classifier

Table 3. Obtained quality metrics of CATDS using existing and proposed classification models.

| Method | Accuracy | Precision | Recall | $F_1$-score | Sensitivity | Specificity |
|---|---|---|---|---|---|---|
| SVM [13] | 70.62 | 79.77 | 66.68 | 64.63 | 100 | 84.21 |
| Random Forest [12] | 84.61 | 88.18 | 82.57 | 83.66 | 100 | 100 |
| TDAS [17] | 86.27 | 89.30 | 84.38 | 84.28 | 100 | 100 |
| SVM-RFE [20] | 90.392 | 90.19 | 86.34 | 86.33 | 100 | 100 |
| Proposed HELM | 92.30 | 92.39 | 92.06 | 92.02 | 100 | 100 |

## 5. Conclusions

This article implemented the CATDS model according to the IAM principles, which classified the healthy, appendicitis, bronchitis, gastritis, heart, and pancreatitis disease classes using HELM classifier. Initially, FNLM is applied on tongue images to improve the digital quality of given image. Next, both texture and color features are extracted from these pre-processed tongue image using color moments and GLCM descriptor. Finally, HELM classifier is used to classify the various diseases using extracted color and texture features. The obtained classification results shown that proposed CATDS using GLCM-based HELM classifier outperformed the existing CATDS models with an increment of 2.11%, 2.43%, 6.62%, and 6.59% in terms of accuracy, precision, recall, and $F_1$-score, respectively. In future, this work can be extended to improve the classification accuracy with deep learning convolutional neural networks.

## Conflicts of interest

The authors declare no conflict of interest.

## Author contributions (Mandatory)

"Conceptualization, SreeRama Prasad Chelluboina; methodology, SreeRama Prasad Chelluboina; software, SreeRama Prasad Chelluboina; validation, SreeRama Prasad Chelluboina; formal analysis, SreeRama Prasad Chelluboina; investigation, SreeRama Prasad Chelluboina; writing—original draft preparation, SreeRama Prasad Chelluboina; writing—review and editing, SreeRama Prasad Chelluboina, Kunjum Nageswara Rao;

## References

[1] D. Shi, C. Tang, S. V. Blackley, L. Wang, J. Yang, Y. He, S. I. Bennett, Y. Xiong, X. Shi, L. Zhou, and D. W. Bates, "An annotated dataset of tongue images supporting geriatric disease diagnosis", *Data in Brief*, Vol. 32, p. 106153, 2020.

[2] Q. Xu, Y. Zeng, W. Tang, W. Peng, T. Xia, Z. Li, F. Teng, W. Li, and J. Guo "Multi-task joint learning model for segmenting and classifying tongue images using a deep neural network", *IEEE Journal of Biomedical and Health Informatics*, Vol. 24, No. 9, pp. 2481-2489, 2020.

[3] E. Gholami and S. R. K. Tabbakh, "Increasing the accuracy in the diagnosis of stomach cancer based on color and lint features of tongue", *Biomedical Signal Processing and Control*, Vol. 69, p. 102782, 2021.

[4] G. Wen, K. Wang, H. Li, Y. Huang, and S. Zhang, "Recommending prescription via tongue

image to assist clinician", *Multimedia Tools and Applications*, Vol. 80, No. 9, pp. 14283-14304, 2021.

[5] D. C. Braz, M. P. Neto, F. M. Shimizu, A. C. Sá, R. S. Lima, A. L. Gobbi, and O. N. Oliveira Jr, "Using machine learning and an electronic tongue for discriminating saliva samples from oral cavity cancer patients and healthy individuals", *Talanta*, Vol. 243, p. 123327, 2022.

[6] S. Dulam, V. Ramesh, and G. Malathi, "Tongue image analysis for COVID-19 diagnosis and disease detection", *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 9, No. 5, pp. 7924-7928, 2020.

[7] J. Ma, G. Wen, C. Wang, and L. Jiang, "Complexity perception classification method for tongue constitution recognition", *Artificial Intelligence in Medicine*, Vol. 96, pp. 123-133, 2019.

[8] Q. Xu, Y. Zeng, W. Tang, W. Peng, T. Xia, Z. Li, F. Teng, W. Li, and J. Guo, "Multi-task joint learning model for segmenting and classifying tongue images using a deep neural network", *IEEE Journal of Biomedical and Health Informatics*, Vol. 24, No. 9, pp. 2481-2489, 2020.

[9] H. Lin, H. Chen, L. Weng, J. Shao, and J. Lin, "Automatic detection of oral cancer in smartphone-based images using deep learning for early diagnosis", *Journal of Biomedical Optics*, Vol. 26, No. 8, 2021.

[10] T. Jiang, X. Guo, L. Tu, Z. Lu, J. Cui, X. Ma, and X. Hu, "Application of computer tongue image analysis technology in the diagnosis of NAFLD", *Computers in Biology and Medicine*, Vol. 135, 2021.

[11] J. Heo, J. H. Lim, H. R. Lee, J. Y. Jang, Y. S. Shin, D. Kim, and C. H. Kim, "Deep learning model for tongue cancer diagnosis using endoscopic images", *Scientific Reports*, Vol. 12, No. 1, pp. 1-10, 2022.

[12] R. O. Alabi, et al., "Comparison of nomogram with machine learning techniques for prediction of overall survival in patients with tongue cancer", *International Journal of Medical Informatics*, Vol. 145, p. 104313, 2021.

[13] J. Shan, et al., "Machine learning predicts lymph node metastasis in early-stage oral tongue squamous cell carcinoma", *Journal of Oral and Maxillofacial Surgery*, Vol. 78, No. 12, pp. 2208-2218, 2020.

[14] R. O. Alabi, et al., "Machine learning application for prediction of locoregional recurrences in early oral tongue cancer: a Web-based prognostic tool", *Virchows Archiv*, Vol. 475, No. 4, pp. 489-497, 2019.

[15] M. Z. Shamim, "Automated detection of oral pre-cancerous tongue images using deep learning for early diagnosis of oral cavity cancer", *The Computer Journal*, Vol. 65, No. 1, pp. 91-104, 2022.

[16] X. Wang, J. Liu, C. Wu, J. Liu, Q. Li, Y. Chen, X. Wang, X. Chen, X. Pang, B. Chang, and J. Lin, "Artificial intelligence in tongue diagnosis: Using deep convolutional neural network for recognizing unhealthy tongue with tooth-mark", *Computational and Structural Biotechnology Journal*, Vol. 18, pp. 973-980, 2020.

[17] J. Li, P. Yuan, X. Hu, J. Huang, L. Cui, J. Cui, J. Xu, "A tongue features fusion approach to predicting prediabetes and diabetes with machine learning", *Journal of Biomedical Informatics*, Vol. 115, p. 103693, 2021.

[18] S. Balasubramaniyan, V. Jeyakumar, and D. S. Nachimuthu, "Panoramic tongue imaging and deep convolutional machine learning model for diabetes diagnosis in humans", *Scientific Reports*, Vol. 12, No. 1, 2022.

[19] R. F. Mansour, M. M. Althobaiti, and A. A. Ashour, "Internet of Things and Synergic Deep Learning Based Biomedical Tongue Color Image Analysis for Disease Diagnosis and Classification", *IEEE Access*, Vol. 9, pp. 94769-94779, 2021.

[20] S. Fan, B. Chen, X. Zhang, X. Hu, and L, Bao, "Machine learning algorithms in classifying TCM tongue features in diabetes mellitus and symptoms of gastric disease", *European Journal of Integrative Medicine*, Vol. 43, p. 101288, 2021.

[21] G. Wen, M. Luo, P. Yang, D. Dai, Z. Yu, C. Wang, and W. Hall, "Fully-channel regional attention network for disease-location recognition with tongue images", *Artificial Intelligence in Medicine*, Vol. 118, p. 102110, 2021.

[22] H. Li, G. Wen, and H. Zeng, "Natural tongue physique identification using hybrid deep learning methods", *Multimedia Tools and Applications*, Vol. 78, No. 6, pp. 6847-6868, 2019.

[23] G. Wen, J. Ma, Y. Hu, H. Li, and L. Jiang, "Grouping attributes zero-shot learning for tongue constitution recognition", *Artificial Intelligence in Medicine*, Vol. 109, p. 101951, 2020.

[24] J. Li, Q. Chen, X. Hu, P. Yuan, L. Cui, L. Tu, J. Cui, "Establishment of non-invasive diabetes risk prediction model based on tongue features and machine learning techniques", *International*

*Journal of Medical Informatics*, Vol. 149, p. 104429, 2021.

[25] S. Balu and V. Jeyakumar, "A study on feature extraction and classification for tongue disease diagnosis", In *Intelligence in Big Data Technologies—Beyond the Hype*, Springer, Singapore, pp. 341-351.