Computers in Biology and Medicine 155 (2023) 106643

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

GeneViT: Gene Vision Transformer with Improved DeepInsight for cancer classification

Madhuri Gokhale [a,b,*], Sraban Kumar Mohanty [b], Aparajita Ojha [b]

[a] *Department of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur, 482001, India*
[b] *Computer Science & Engineering, PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, 482005, India*

ARTICLE INFO

Keywords:
Gene expression data
Cancer classification
Vision transformer
Deep learning

ABSTRACT

Analysis of gene expression data is crucial for disease prognosis and diagnosis. Gene expression data has high redundancy and noise that brings challenges in extracting disease information. Over the past decade, several conventional machine learning and deep learning models have been developed for classification of diseases using gene expressions. In recent years, vision transformer networks have shown promising performance in many fields due to their powerful attention mechanism that provides a better insight into the data characteristics. However, these network models have not been explored for gene expression analysis. In this paper, a method for classifying cancerous gene expression is presented that uses a Vision transformer. The proposed method first performs dimensionality reduction using a stacked autoencoder followed by an Improved DeepInsight algorithm that converts the data into image format. The data is then fed to the vision transformer for building the classification model. Performance of the proposed classification model is evaluated on ten benchmark datasets having binary classes or multiple classes. Its performance is also compared with nine existing classification models. The experimental results demonstrate that the proposed model outperforms existing methods. The t-SNE plots demonstrate the distinctive feature learning property of the model.

1. Introduction

The Bioinformatics [1] field has become prominent in handling biological issues with computational mathematics and machine learning techniques. These techniques often require large amounts of data generated by different systems and devices. Bioinformatics has been extensively applied in different fields involving biology and medicine such as drug development, genomics, transcriptomics, evolutionary biology, proteomics, genomics, precision medicine, and population genetics [2,3]. These fields of study help acquire better knowledge of intricate biological systems [4,5]. Proteomics deals with the study of proteins, and genomics defines the study of genome. The genome is usually developed with Deoxyribonucleic acid (DNA) sequences [6] which comprise a set of genes that transmit genetic information from parent to offspring and the transcripts that contain ribonucleic acid (RNA) copies [7].

The DNA microarray data is obtained from the cell and tissue samples, and evaluation of such data has become increasingly significant in recent years. The fundamental achievement of microarray technology is its capacity to record the expression levels of thousands of genes simultaneously in a single experiment. To convert gene expression data into useful information, efficient and effective computational techniques are being developed. These techniques include data mining, machine learning, and statistical analysis. However, researchers encounter potential barriers in exposing new and important information from gene expression or microarray data due to various reasons. The curse of dimensionality, imbalanced data, missing values, noisy data, redundant genes, and bias are some of the common issues with gene expression data. Many feature selection and dimension reduction methods combined with classification approaches are used in analyzing microarray data [8].

In the last two decades, a plethora of Machine Learning (ML) approaches have been developed for gene expression analysis and cancer categorization [9–14]. However, these methods have certain limitations in processing high dimensional data, making the classification of cancerous genes a challenging task [15]. Even with dimensionality reduction techniques, these methods do not perform very well. It is well known that the feature selection methods used in conjunction with ML approaches face the peaking phenomenon that shows that the error of a classifier initially decreases with the increasing number of features and then suddenly starts shooting up as the number of features grows.

[*] Corresponding author at: Computer Science & Engineering, PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, 482005, India.

E-mail addresses: madhurigokhale@iiitdmj.ac.in (M. Gokhale), sraban@iiitdmj.ac.in (S.K. Mohanty), aojha@iiitdmj.ac.in (A. Ojha).

https://doi.org/10.1016/j.compbiomed.2023.106643
Received 14 September 2022; Received in revised form 3 January 2023; Accepted 5 February 2023
Available online 6 February 2023

Deep learning (DL) techniques have an advantage over traditional ML methods in that they automatically extract meaningful features and learn hidden representations from the data. Deep learning algorithms have become powerful tools in computer vision, pattern recognition, natural language processing, and in biomedical applications including genomics [16–19]. Some of the commonly used deep neural architectures include Convolutional Neural Networks (CNN), Autoencoders (AE), Generative Adversarial Networks (GAN), and Recurrent Neural Networks (RNN) [20].

Since the gene expression data is high dimensional, many researchers have explored deep learning techniques for cancer classification and these techniques have shown promising performance. Recent works on gene expression analysis have also combined the feature learning capabilities of deep neural networks with traditional ML classifiers to improve the performance of their cancer detection models. Stacked autoencoders, known for their excellent feature extraction capabilities, have been successfully applied with various ML and the CNN based classifiers for building effective gene expression based cancer detection models.

In recent years, transformer networks have emerged as powerful models in a variety of artificial intelligence tasks [21]. In computer vision, vision transformer (ViT) network has gained much popularity due to its significantly improved performance in different applications. The architecture of the ViT is inspired by the transformer network [21]; it employs self-attention layers to preserve long-term dependencies and can capture extremely effective and complex relationships between spatial positions. The multilayer perceptron (MLP) present in the architecture further facilitate the ViT in learning more generic and flexible correlations across spatial locations from raw data [22]. As a result, ViT models have pushed the state-of-the-art in a variety of vision tasks, including image classification [23], object recognition [24], semantic segmentation [25] image colorization [26], low-level vision [27], and video comprehension [28], to mention a few. Furthermore, current research suggests that the prediction errors of ViT models have greater similarity with human errors than with those of CNNs [29–32]. These important characteristics of ViT have piqued the interest of medical community in adapting them for biomedical applications [33].

The ViT uses data in image format but the gene expression data is sequential in nature. To utilize ViT for the gene expression analysis and cancer classification task, an improved version of the original DeepInsight method is introduced in the present paper that organizes the gene expression data using semantic similarity. The DeepInsight method has been proposed by Sharma et al. [34] to convert non image data to an image format using t-Distributed Stochastic Neighbor Embedding (t-SNE) visualization of input features. The idea of using the DeepInsight comes from the fact that the microarray data is dispersed throughout a high-dimensional data space, making it difficult to distinguish between phenotypes. Therefore, it is crucial to arrange components in a way that allows important characteristics to be extracted for analysis. The DeepInsight maps the data features into a 2D space after transposing the input data matrix to create an image that displays feature similarities. The new input data points are then represented as pixels on the resultant 2D image frame and the images are categorized using a CNN. However, if more than one feature acquire the same pixel location in the frame, the features are averaged and placed in the location during feature mapping. This is a significant limitation of the approach as the process of averaging result in loss of information. To overcome this issue, an Improved DeepInsight method with channel expansion is proposed in the present work that helps in better performance of the model. Main highlights of the present work are as follows.

- One of the first gene selection and cancer classification method that employs a vision transformer on a variety of genes data and cancer types.
- The proposed method uses a stacked autoencoder for dimensionality reduction while extracting the most relevant gene combination as the latent representation of the data.
- An Improved DeepInsight method is proposed that works on the principle of t-Distributed Stochastic Neighbor Embedding (t-SNE) and a channel expansion algorithm for preserving all the relevant genes selected by the stacked autoencoder so that the data can be served to the vision transformer in image format.
- The proposed method outperforms nine state-of-the-art gene selection and cancer classification methods on ten benchmark datasets.
- The t-SNE plots demonstrate the distinctive feature learning property of the model.

The rest of the paper is organized as follows. An overview of the state-of-the-art methods in gene selection and classification is presented in Section 2. The proposed model of gene selection and cancer classification is discussed in Section 3. Experimental findings using ten benchmark gene expression datasets are discussed in Section 4. Section 5 concludes with a summary of the work and its future scope.

2. Related work

Numerous feature selection, dimensionality reduction, and classification algorithms have been proposed in the literature to improve microarray gene expression analysis for better data classification performance [35]. Since the data comprises gene expression levels for over 20 thousand genes, traditional ML algorithms struggle to perform well owing to the curse of dimensionality [36]. With the remarkable performance of DL algorithms on high-dimensional data, recent research is focused on using deep neural network architectures, that have demonstrated higher precision and accuracy in cancer detection through gene selection and classification. In this section, an overview of recent approaches for gene selection and cancer classification is given, relevant to the present work and summarized in Table 1.

Liu et al. [37] presented a 1D CNN with sample expansion technique to carry out the classification process of cancer gene expression data. They analyzed both the Sample Expansion-Based 1DCNN (SE1DCNN) and Expansion-Based SAE (SESAE) to build an effective gene expression based cancer classifier model. Zeebaree et al. [38] developed a DL approach based on CNN for microarray data classification. The CNN model obtained 97.62% classification accuracy for brain dataset while 91.86% accuracy was attained for the prostate cancer data. The error rate was relatively high and the model also required high memory. Maniruzzaman et al. [12] developed a system that addressed the fundamental problem of identification of risky differential genes through statistical tests and established an ML strategy for cancerous gene prediction. Their model exhibited an average accuracy of 90.50% but the convergence was very slow.

Adem et al. [39] developed a hybrid approach of using stacked autoencoder with k-nearest neighbor (KNN) algorithm for the determination of breast cancer disease using microarray analysis. But the classification accuracy was not very impressive in their model. Shah et al. [40] also proposed a hybrid DL approach using Laplacian Score and a Convolutional Neural Network (LS-CNN) model for classification of microarray data. In this method most relevant gene are selected from high dimensional data using Laplacian score and then classified the selected genes by modified 1DCNN model. Although in comparison the proposed method showed better accuracy on binary datasets but less accurate on multiclass datasets. Kilicarslan et al. [41] proposed a combination of Relief and stacked autoencoder methods for dimensionality reduction along with SVM and CNN for classification. However proposed method was applied on binary datasets only, handling multiclass datasets need to be explored.

Debeta et al. [42] presented a kernel-based Fisher score (KFS) method for extracting relevant genes. For the classification of high dimensional microarray gene expression data, an enhanced chaotic Jaya (CJaya) procedure optimized with CNN model (CJaya-CNN) was adopted. Systematic handling of larger datasets may improve accuracy and achieve required robustness. Deng et al. [43] introduced

Table 1
Characteristics of existing approaches for gene selection and cancer classification.

Author	Technique used	Objective	Performance (%)
Traditional Machine Learning Approaches			
Dwivedi et al. [11] 2018	ANN	To classify acute myeloid leukemia and acute lymphoblastic leukemia cancer.	Acc—98
Maniruzzaman et al. [12] 2019	Ten ML methods	To determine risky differential genes.	Acc—90.50
Houssein et al. [13] 2021	BMO and SVM	To select most appropriate genes.	Avg SD −0.35
Reddy et al. [14] 2022	SVM	To increase the genetic data potentiality.	Pre—93.64
Deep Learning Approaches			
Liu et al. [37] 2017	Inf-FS, SE1DCNN and SESAE	To mitigate the sample insufficiency issue of gene expression data.	Acc—95.33
Zeebaree et al. [38] 2018	RF, CNN	Classification of microarray data with DL methods.	Acc—97.62
Adem et al. [39] 2020	SAE and KNN	To diagnose breast cancer using microarray data.	Acc—91.24
Shah et al. [40] 2020	LS-CNN	To develop hybrid deep learning model for classifying cancer. data	Acc—97
Kilicarslan et al. [41] 2020	SVM, CNN	To develop a hybrid methodology for reducing the data dimensionality.	Acc—99.86
Debeta et al. [42] 2021	KFS, CNN	To develop a model for extracting relevant genes.	Acc—98.2
Deng et al. [43] 2022	XGBoost, MOGA	To classify cancer disease using two stage gene selection method.	Acc—90.24
Nature Inspired Approaches			
Dabba Ali et al. [44] 2021	MIM-mMFA +SVM	To reduce the feature dimensionality.	Acc—96
Ab et al. [45] 2021	PSO+SVM	To classify the cancer data.	Acc—96.15
Maulidina et al. [46] 2021	PSO-GA-SVM	To optimize the network parameters.	Acc—97.69
Sree et al. [47] 2022	GWO and RF	To classify colon cancer data.	Acc—95.16
Seetharaman et al. [48] 2022	BBA and SVM	To diagnose cancer using microarray datasets.	Acc—95.85

Acc—Accuracy, Avg SD—Average Standard Deviation, Pre—Precision, FS—Feature Selection

a two stage gene selection method through the integration of extreme gradient boosting (XGBoost) and multi-objective optimization genetic algorithm (MOGA) for effective cancer classification. Although the feature selection approach was novel, the overall classification performance was not at par with the prevailing methods.

Ab et al. [45] presented an effective approach for cancer classification through the adoption of Particle Swarm Optimization (PSO) and Support Vector Machine (SVM). The PSO approach was employed to choose the optimal relevant features and kernel attributes. Their model was investigated over breast cancer datasets and exhibited good performance. Maulidina et al. [46] also used a combination of the PSO with Genetic Algorithm (GA) and SVM (PSO-GA-SVM) to build a cancer classification model using gene expression data. The PSO and GA were employed to optimize the network parameters of the SVM model. Their approach led to an accuracy of 97.69%, precision of 98.46%, recall of 98.82% and F1-score of 97.66% on a lung cancer dataset.

Sree et al. [47] utilized a data mining policy and an optimized feature selection approach with a limited dense tree forest as a colon cancer classifier. The combination of gain information and feature selection outcomes using gray wolf optimizer (GWO) were fed as the input to the random forest classifier. The model attained an accuracy of 95.16%. Seetharaman et al. [48] introduced the correlation feature selection filter and binary bat algorithm (BBA) in association with greedy crossover. Through the correlation founded feature selection filter, the gene feature subsets were obtained. The optimization of features was carried out through BBA whereas the sub optimal solutions attained by BBA pre-convergence were reset through the greedy crossover. The SVM was used as the classifier for the microarray datasets. The features were highly reduced through multi-objective solutions and the classification accuracy of 95.85% was attained.

Lu et al. [49] have proposed a two-stage approach that includes contrastive pre-training on glioma sub-type categorization in the brain, followed by feature aggregation using a transformer-based sparse attention module. Gheflati et al. [50] have compared the performance of pure and hybrid pre-trained ViT models for the classification of breast cancer using ultrasound images. Experiments on two breast ultrasound datasets given by Al-Dhabyani et al. [51] and Yap et al. [52] reveal that ViT-based models outperform CNNs in image classification into benign, malignant, and normal categories. Khan et al. [53] have proposed a Gene-Transformer model for predicting lung cancer subtypes. Experiments on the TCGA dataset show that the Gene Transformer outperforms CNN baselines. Chen et al. [54] have also proposed a multi-scale GasHis-Transformer model for detecting stomach cancer. Jiang et al. [55] have presented a ViT-CNN ensemble model for diagnosing acute lymphocytic leukemia.

Motivated by the performance and behavior of ViT, an end-to-end solution for cancer classification utilizing gene expression is proposed in the present paper that leverages self-attention based feature extraction and classification capability of ViT. The proposed model uses a stacked autoencoder for gene selection, an Improved DeepInsight method for conversion of gene expression data into image format by grouping similar genes into a 2D space [34], and the ViT for cancer classification. Multi-head self-attention module allows the DL model to learn complicated genomic information from the gene expression data across various cancer types and subtypes. Head cooperation improves generalizability over large datasets and results in superior performance for binary and multiclass cancer identification and classification task.

3. Proposed method

In this section, we provide a new approach to cancer classification based on a hybrid deep learning strategy that combines a stacked autoencoder with the vision transformer. The framework of the proposed technique consists of three phases—data preprocessing, conversion of the data to image format using an Improved DeepInsight approach, and a vision transformer for cancer classification. In the preprocessing phase, the microarray data is augmented by adding Gaussian noise to reduce the class imbalance and to increase the dataset size. Next, the augmented data is normalized using Min–Max normalization, and then a stacked autoencoder is used to reduce the dimension by filtering out irrelevant genes. In the data conversion phase, samples are processed to create gene expression images using an Improved DeepInsight approach with channel expansion. Finally, in the classification phase, the vision transformer is trained to classify the samples into the desired number of categories. Fig. 1 illustrates the schematic block diagram of the proposed method.

3.1. Preprocessing

The data preprocessing is needed for gene expression datasets, as the imbalanced microarray biological data contains a substantial amount of noise and bias. Further, the datasets are generally high dimensional but are having small number of samples. The preprocessing methods help alleviate these problems to a great extent. In the present method, this phase consists of three steps; Step 1—augmentation of microarray

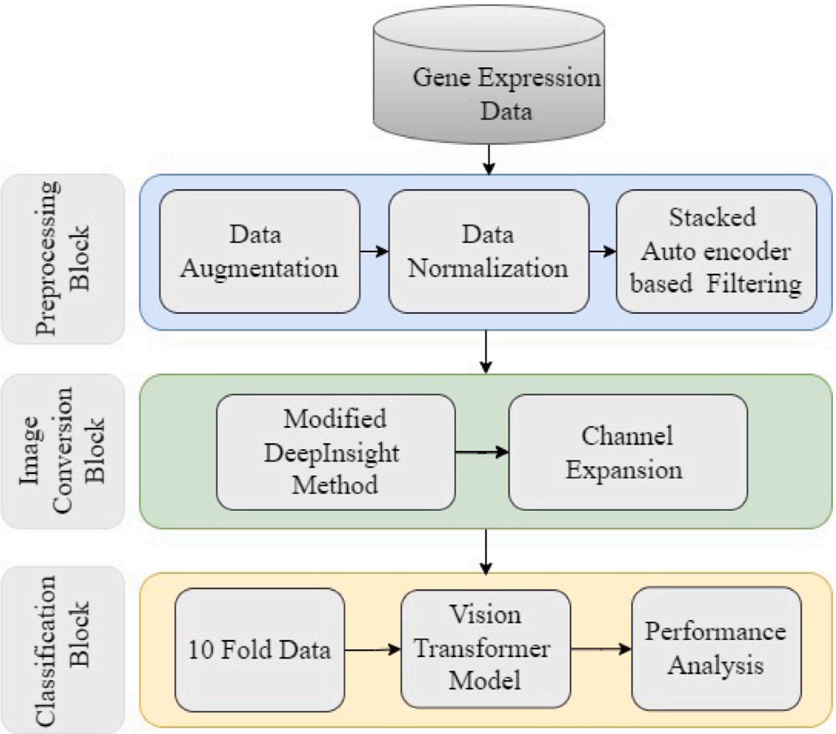


Fig. 1. Schematic block diagram of the proposed method.

dataset, Step 2—normalize the dataset, and Step 3—apply a stacked autoencoder to reduce the number of genes and extract relevant genes. The next section covers these three steps in details.

3.1.1. *Data augmentation*

The distribution of gene expression data is intrinsically class imbalanced. Data augmentation is often a standard practice to increase the number of samples and reduce the class imbalance in the gene expression data [56]. In this work, data augmentation is performed by inserting a low magnitude Gaussian noise into the data samples. Gaussian noise is a statistical noise randomly taken from the normal distribution, also known as Gaussian distribution. It is defined as follows.

$$N(X) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2 / 2\sigma^2}, \quad (1)$$

where $\mu = 0$ is the sample mean and $\sigma = 1$ is the standard deviation. One of the primary motivations in using Gaussian distribution is that all random variables tend to be Gaussian in nature. In this study, data samples are subjected to Gaussian noise with a noise factor of 0.1 in order to produce new samples. For this, $10 - 20\%$ of samples from each class are randomly selected, and a random noise sample is created using Eq. (1) with a mean equal to the sample mean and a standard deviation equal to 1. The selected samples are then subjected to the random noise to produce more samples. These additional samples are mixed with the original samples. It is worth mentioning that the noise in the data may affect its features. Therefore, the noise should be so added that the nature of the augmented data stays unchanged after augmentation.

Before using the augmented data, it was necessary to check if noise used in the data augmentation had affected the nature of samples. Accordingly, an experiment was performed using Pearson correlation coefficient to verify how correlated were the genes in the datasets before and after data augmentation. The correlation coefficient value of 1 was observed between the sets of data samples without noise and the sets of the same samples with added noise. Another experiment was performed by randomly selecting 20 samples from each class of the datasets before and after data augmentation. Pearson correlation coefficients were computed between the two sets of each of the datasets. The average Pearson correlation coefficient was computed by repeating the process ten times. The Pearson correlation coefficient for all datasets in each of the experiments was found to be more than 0.7. This helped in establishing the fact that the addition of random noise did not significantly alter the data characteristics.

3.1.2. *Data normalization*

After the data augmentation, a data normalization process is also performed in the present method. DNA microarray technology allows thousands of genes to be examined at once. However these collected samples are measured at various scales and therefore may not contribute evenly to the model building, and may result in bias. For such type of data, normalization approaches try to remove systematic experimental bias and technical variance while maintaining the biological variation. Gene-wise normalizing has become a standard procedure in the study of gene expression. In the present work, Min–Max scaling is used to normalize samples of all the datasets. Min–Max scaling is performed using Eq. (2).

$$X_{Norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

where X_{min} and X_{max} are the minimum and maximum values of a gene in the dataset.

The microarray data utilized in this study is an $M \times N-$ dimensional array, with each row having a single sample and each column having a single gene (feature). Each individual gene is subjected to a linear transformation in the Min–Max normalization process that maintains the relationship with the original data values. Min–Max normalization ensures the stability of the weight and bias convergence process [57].

3.1.3. *Stacked autoencoder based gene filtering*

In gene expression based cancer prediction models, autoencoders have been extensively applied for gene filtering or selection. In the present work, a stacked autoencoder is used for filtering the gene expressions and reducing the dimensionality. An autoencoder is an

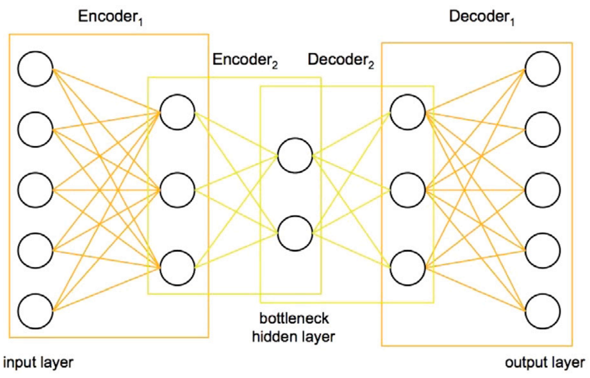


Fig. 2. A Stacked Autoencoder Model.

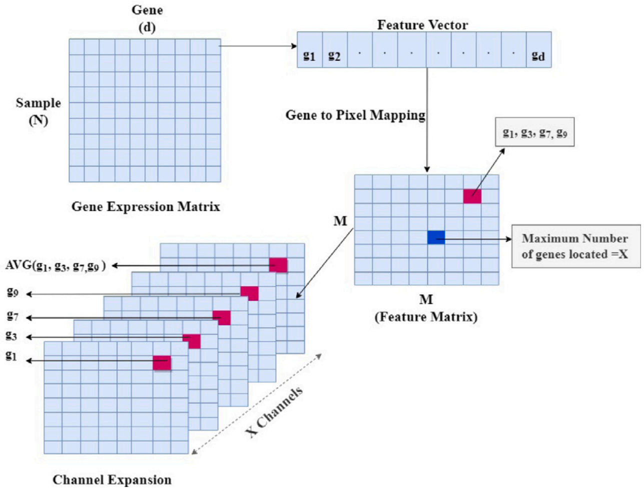


Fig. 3. Channel expansion in Improved DeepInsight method.

artificial neural network mainly consisting of two components: an encoder and a decoder as illustrated in Fig. 2. The encoder learns the representation of input samples and generally transforms it to a low-dimensional vector known as its "latent vector”, while the decoder learns to reconstruct the input from the latent vector [58,59]. In this work, a stacked autoencoder is used for dimensionality reduction by filtering out irrelevant genes. The architecture of the stacked autoencoder is based on an earlier work done by the authors [60]. The advantages of this architecture is that it provides the highest accuracy for all the ten datasets. Further, it has been investigated in [60] that the genes selected for data representation using the proposed stacked autoencoder are biologically significant, and are the biomarkers for cancer types and subtypes. The stacked autoencoder has an input layer, two hidden layers, a coding layer in its encoder part, and then two hidden layers and the output layer for the decoder. For the dimensionality reduction, the size of the data samples is gradually reduced by selecting the number of neurons in each layer as follows. The first hidden layer has 90% neurons, the second hidden layer has 75% neurons, and the code layer's size is lowered to 50% of input layer.

The cross entropy loss function is used to build the gene filtering stacked autoencoder model as given in Eq. (3).

$$Loss = \frac{1}{m}\Sigma_1^m(X_i \log(X_i') + (1 - X_i)(\log(1 - X_i'))), \tag{3}$$

where X_i denotes the input to the stacked autoencoder and X_i' denotes the decoder's reconstructed input ($i = 1, \dots, m$). The average is taken over all the m samples in a batch. Standard optimization method ‘ADAM’ [61] is used to optimize the loss.

3.2. Improved DeepInsight with channel expansion

Sharma et al. [34] have recently proposed the DeepInsight method for transforming non-image data into images to utilize the capabilities of CNN architecture in feature extraction and classification tasks. When the dimensionality of a data sample is very high, fitting all of the elements (genes) into a frame of size $M \times N$ may result in a large image size. And if the frame size is chosen to be smaller, one needs to compress the elements in such a way that the data characteristics are preserved. Instead of using domain-specific information to rearrange input feature vectors, Sharma et al. [34] have adopted a broader approach that incorporates an initial kernel principal component analysis (k-PCA) [62] or t-SNE [63] to transform the input feature into a $2D$ feature frame (image). In the next step, the convex hull technique is applied to predict the smallest rectangle that includes significant features. To align the image with the $x - y$ frame of the Cartesian coordinate system, rotation with a desired angle is applied on the frame, and then Cartesian coordinates are translated to pixel coordinates. Following that, element values are mapped to pixel positions to create an image of a feature vector. The method is shown to perform well in conjunction with CNN for classification of gene expression data [34]. One disadvantage of the method is that several features of an input vector are mapped onto a single pixel location and their values are averaged to generate a single pixel value. This is a lossy compression and can cause under-representation of various genes that might be similar in nature, but have their own biological significance in the data representation. To cope with this issue, an Improved DeepInsight method for the conversion of non-image data into an image format is proposed in the present paper that preserves the feature information independently rather than averaging them. The Improved DeepInsight uses a channel expansion process instead of averaging the feature values. In the following, the channel expansion process is described.

Suppose a pixel frame size of $M \times N$ is used to convert the gene expression data into an image. The latent vector obtained through the trained stacked autoencoder is passed through the process of frame construction as detailed in the previous paragraph. Note that multiple features may be assigned a single pixel position. Let $k_{i,j}$ number of features be assigned to a pixel position (i, j). Consider the maximum of $k_{i,j}$ taken over all values of $i = 1, \dots, M$; $j = 1, \dots, N$. Let this maximum value be k. Then the pixel frame is expanded to k channels, and in each channel one feature value is assigned at the pixel position (i, j). If $k_{i,j} < k$ for some pixel position (i, j), then the first $k_{i,j}$ channels are filled with the feature values assigned to the pixel position, and the remaining $k - k_{i,j}$ channels are filled with the average of the first $k_{i,j}$ values. In this way the pixel frame is transformed to a $M \times N \times k$ volume. An example of channel expansion process in Improved DeepInsight is presented in Fig. 3. The reason for calling it improved DeepInsight is that the performance of the classifier improves if the Improved DeepInsight is used in place of the original DeepInsight. This is illustrated later in Section 4.5.

Once the data is transformed to image form, it is fed to the ViT, that is trained to classify the data.

3.3. GeneViT: Gene vision transformer for cancer classification

The Vision Transformer (ViT) [64] is a pure transformer network that has emerged as a powerful network for extracting image features and performing classification task through its self-attention mechanism. The architecture of a GeneViT model is depicted in Fig. 4. The self-attention blocks combined with multilayer perceptron (MLP) help the network in extracting most relevant global features. First an image is split into non-overlapping fixed size patches that are subjected to linear projection and positional embedding process. Positional embedding preserves the relative position of a patch with respect to the input image. After this, the flattened patches are passed through a stack of T number of transformer blocks. The main components of a typical

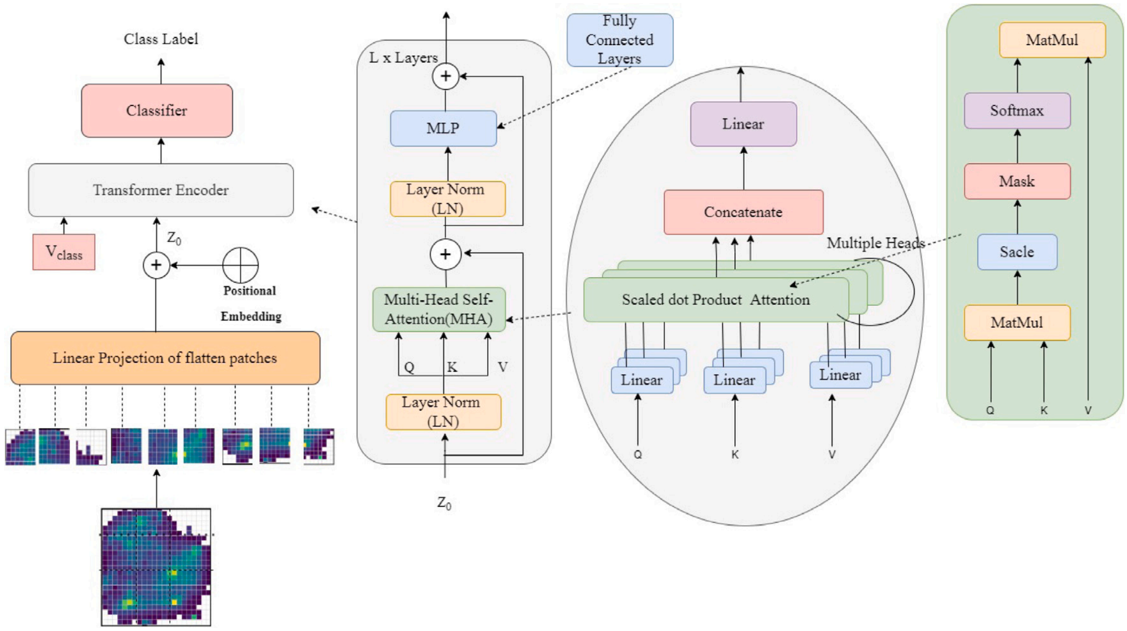


Fig. 4. Architecture of the proposed GeneViT.

Algorithm 1 Pseudo code of gene selection and classification method

```
Require: Training dataset
Ensure: Classification result
  for each dataset $D : \{D_1, D_2, D_3 \dots\dots\dots\dots D_N\}$ do
      Step 1: Get prepossessed gene dataset:
      1: Data augmentation
      2: Normalization
      3: Gene filtering by the stacked autoencoder
      Step 2: Convert each latent vector representation of the data into
  a 2D array using the DeepInsight
      1: Apply t-SNE to the latent vectors obtained from the encoder.
      2: Apply convex hull algorithm.
      3: Apply rotation to fit the projected data in the smallest 2D (
  rectangular) array to convert the data into image format.
      4: Return the 2D array with no of feature values at each location.
      Step 3: Get the converted data sample in the image format using
  Improved DeepInsight method with channel-wise expansion
      1: Set the number of channels as the maximum number of features
  ($K^*$ ) at any given location of the array.
      2: Expand the values into different channels keeping one feature
  value at each channel.
      3: Compute the average of each row
      4: Set average value of each row at each empty location,
      5:For each column in the matrix, generate the image
      Step 4: Classification by GeneViT Model
  end for
```

transformer block are multi-head self-attention (MHA) and MLP. Each one is preceded by a normalization layer and residual connection at the end.

In each block of MHA an input vector is converted into three distinct vectors: k- key; q- query; and v- value. In the next step, the vectors generated from various inputs are packed into three individual matrices, computed as $Q = XW_Q, K = XW_K$, and $V = VW_V$; where W_Q, W_K, and W_V are the weight matrices. A dot-product of Q and K is taken to generate a score matrix based on the saliency of the embedded patch. Then, the SoftMax activation function is applied to the score matrix. Consequently, the weighted sum values are calculated with the following equation:

$$Attention(Q,K,V) = Softmax(\frac{Q.K^T}{\sqrt{d_k}}).V \tag{4}$$

where d_k represents the dimension of the vector K. Finally, self-attention matrices are merged and sent to a linear layer and the regression head. Self-attention facilitates the categorization of relevant semantic information at image locations. In the transformer encoder, often known as MHA, there may be any number of self-attentions. Eq. (5) is used to calculate the MHA block's output.

$$MHA_{out} = MHA(NORM(x_{in})) + x_{in}, \tag{5}$$

where MHA_{out} is the output of multi-head self-attention layer, MHA is multi-head self-attention layer, NORM is the normalization layer, and x_{in} is the input to the transformer block

In the transformer block, MLP is layered after the MHA layer. In MLP, GeLU activation function is used. To calculate the GeLU activation, the input is multiplied by its Bernoulli distribution. As shown in Fig. 4, the transformer block contains skip connections from the MHA output. The output of the transformer block can be calculated using Eq. (6).

$$TF_{out} = MLP(NORM(MHA_{out})) + MHA_{out} \tag{6}$$

where MLP is the multi layer perceptron block, and TF_{out} is the output of the transformer block.

In the present work the fundamental settings of ViT are used to build the proposed GeneViT model. First, a series of non-overlapping image patches is created from the given input with an 8 × 8 patch size as opposed to the 16 × 16 patch size used in the original ViT model. Then two MLP layers with 2096 and 1048 neurons are used. Based on an ablation study, the number of transformer blocks is set to 4, and the number of heads is set to 8. Other hyperparameters are mentioned in Table 3. Section 4.4 summarizes the results of the ablation study for setting up the hyper-parameters pertinent to the proposed GeneViT model. Steps of the proposed method are summarized in Algorithm 1.

4. Experimental results and discussion

A GeneViT cancer gene expression data classification model is developed using ten publicly available binary and multi-class high-dimensional microarray datasets. To finalize the model architecture

Table 2
Datasets used in experiment.

S No	Dataset Name	Genes	Classes	Samples	Class Distribution	Augmented data Sample	Augmented data Distribution
DS1	Colon	2000	2	62	40/22	122	60/62
DS2	Breast Cancer	24481	2	97	51/46	194	97/97
DS3	Central Nervous System	7129	2	60	39/21	155	58/57
DS4	ALL-AML	7129	2	72	25/47	144	72/72
DS5	Ovarian	15154	2	253	162/91	344	172/172
DS6	ALL-AML (Leukemia-3c)	7129	3	72	38/9/35	144	48/48/48
DS7	MLL	12582	3	72	24/20/28	144	48/48/48
DS8	ALL-AML (Leukemia-4c)	7129	4	72	39/9/21/4	144	43/34/34/33
DS9	SRBCT	2308	4	83	29/11/18/25	169	44/40/40/45
DS10	Lung	12600	5	203	139/17/6/21/20	331	139/51/18/63/60

and gauge its effectiveness, a comprehensive trial of experiments was performed. Additionally, the effectiveness of the model was evaluated against nine recent methods using traditional ML and current DL techniques. The following sections present the details of experimental datasets, evaluation parameters, experimental setup, and performance analysis of the proposed model, and a comparative analysis of the model's performance with other competing methods. Furthermore, the t-SNE approach is employed to illustrate the effectiveness of GeneViT in extracting discriminating features of samples for cancer classification.

4.1. Model building

For each datasets, first the data prepossessing is performed and then the samples are passed through the Improved DeepInsight method to reshape each input sample into a $64 \times 64 \times c$ array. The proposed GeneViT model takes an input image of size $64 \times 64 \times c$. Each of the input image is first split into non-overlapping patches of size 8 × 8 as shown in Fig. 4. The flattened patches are then passed through the linear projection block. After the linear projection, a feature vector of size $4096 \times c$ is generated. This vector is fed to a stack of four transformer blocks for features extraction. The output of the last transformer block is flattened to make a 1-dimensional vector. At the end, a fully connected layer with softmax activation is added, with the number of neurons equal to the number of classes in the dataset. The proposed GeneViT model is trained and tested on a variety of datasets using 10-fold cross validation. The gene expression dataset is randomly split into 10 identical sub-datasets, nine of which are utilized for training and the remaining one for testing. The advantage of this strategy is that it is unconcerned with how the samples are partitioned. Each sample appears once in a test set and nine times in the training sets. For training the model, categorical cross-entropy loss with 'AdamW' optimizer is used with a maximum of 100 epochs. The learning rate is set to 0.0001 with a weight decay of 0.001 and the batch size of 30. The average performance of the cancer classification model is assessed after the training process is completed. The experimental results on all the datasets are presented in Section 4.5.

4.2. Experimental datasets

For the experimental analysis, ten different types of benchmark datasets are utilized that have been used in the previous research works and are diversified in nature. The datasets are obtained from http://csse.szu.edu.cn/staff/zhuzx/Datasets.html [65]. Although most of the microarray datasets are binary in nature, some multi-class datasets have also been taken up in the present study for a better understanding of the outcomes. Table 2 gives the details of the datasets such as the number of genes, classes, samples with class distribution, numbers after the data augmentation, and their class distribution. In the following, we briefly describe each dataset.

Table 3
Hyperparameters of GeneViT.

Hyper-parameter	Model	
	Stacked Autoencoder	Vision Transformer
Batch size	30	30
Loss function	Cross entropy	Cross entropy
Optimizer	Adam	AdamW
Learning rate	0.001	0.0001
Dropout rate	–	0.1
Number of heads	–	8
Patch size	–	8 × 8
Projection Dimension	–	64
No of transformer layers	–	4
Epochs	100	100

The colon cancer dataset contains 62 samples with 40 tumor biopsies from colon adenocarcinoma specimens and 22 from normal parts of the colons of the same patients. In each sample, there are 2000 genes [65]. The breast cancer dataset consists of 97 samples with 24481 genes in each sample [39]. There are 46 samples of patients with a history of metastasis with samples taken in a span of 5 years of the initial diagnosis. It also contains 51 samples of patients without any metastasis. The Central Nervous System (CNS) dataset used in the present study has 60 samples with 7129 genes in each sample. These samples are taken from the tumor causing tissues found in the central nervous system of the body [65].

Leukemia is another group of datasets used in this work [65]. Leukemia is a primary bone marrow disorder. The datasets having samples under different categories of Leukemia. Each dataset contains 72 samples with 7129 genes. The samples are of different cancer categories. In the present work, three datasets from Leukemia group are considered, containing 2 classes, 3 classes and 4 classes of Leukemia (Please refer Table 2, DS4, DS6, DS8). Ovarian cancer is caused due to uncontrollable growth of cells in the ovaries. The lump of tissues are produced due to abnormal growth. It is the most common type of cancer in women. The Ovarian cancer dataset contains 253 samples with 15154 genes (91 cancerous and 162 normal) [65].

In the present experiments, another Leukemia dataset is used, namely—mixed-lineage leukemia (MLL) [66] that is frequently engaged in translocation-associated gene fusion events in children leukemia. A total of 12,582 genes and the three cancer types : acute lymphoblastic leukemia (ALL), myeloid lymphoid leukemia (MLL), or acute myeloid leukemia are used to characterize each of the 72 samples in this dataset.

The Small-Blue-Round-Cell Tumor (SRBCT) are tumors that grow in the abdomen and pelvic area of the body. The SRBCT dataset includes information on four cancer types—rhabdomyosarcoma, non-Hodgkin lymphoma, neuroblastoma, and Ewing sarcoma (RMS). In this dataset, 83 samples and 2308 genes in total are present [67]. We have also used Lung cancer dataset for the present investigation. Lung cancer is the most common cancer in humans that is characterized by uncontrolled cell growth in the lung tissues. The dataset used in this study contains 203 tissue samples, each described by 12600 genes in 5 categories [65].

4.3. Experimental setup and evaluation parameters

The proposed model and all the models under comparison were implemented on Keras platform with TensorFlow backend support. Python language and "Scikit-learn" library were used for coding the models. All tests were performed on a PC with a 5 GHz Intel Core i7 processor and 8 GB of RAM.

Performance of the proposed model GeneViT was evaluated using the standard classification measures : Accuracy, Precision, Recall (Sensitivity), F1-score, Kappa score, Area under curve, Specificity and the Confusion matrix [68]. The average values were derived from 10-fold cross-validation. The standard definitions of measures are given below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN} \qquad (Sensitivity)$$

$$Specificity = \frac{TN}{TN + FP}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$Kappa = \frac{P_o - P_e}{1 - P_e}$$

$$AUC = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right)$$

Here TP, TN, FP, and FN denote the number of true positive, true negative, false positive, and false-negative classifications respectively. In the Kappa score, P_o and P_e refer to the observed and expected probabilities respectively. Due to biomedical data, sensitivity and specificity are also applied for the result analysis. In a medical test, a high sensitivity in the model's prediction ensures that a positive is positive, while a high specificity indicates that it is less often that the model will incorrectly diagnose a positive when it is actually not positive. The sensitivity is also termed as recall in the literature.

Furthermore, confusion matrix, receiver-operating characteristic (ROC) curve, and t-SNE plot are used to assess the performance of the GeneViT classifier. Additionally, a t-test is conducted for establishing that results of the proposed approach are statistically significant as compared to those of the other methods.

4.4. Ablation study

To build the proposed cancer classification model, an ablation study was conducted with a comprehensive set of experiments keeping three main components of the model in mind. First, the dimensionality reduction/gene selection methods were analyzed with a basic structure of the ViT as the classifier, while using DeepInsight and Improved DeepInsight one by one. Six frequently used dimensionality reduction methods, namely—Chi-Square test (CS), Mutual Information(MI), ReliefF, minimum redundancy maximum relevance (mRmR), Principal Component Analysis (PCA), and the stacked autoencoder were investigated in the experiments on all the ten datasets. The experiments were also repeated with 4 and 8 transformer blocks, each with 4 and 8 attention heads in the ViT. Then the most suitable dimensionality reduction method and GeneViT architecture were finalized based on the experimental results. It was observed that the Improved DeepInsight was performing better than the original version of DeepInsight.

For the sake of brevity, only a selective set of experimental results are presented here. More precisely, results of the ablation study are presented for all the six dimensionality reduction methods with the best performing GeneViT structure having 4 transformer blocks, each having 8 attention heads. In this experiment, stacked autoencoder turns out to be the best model for gene selection. Then the results of experiments with stacked autoencoder, with DeepInsight and Improved DeepInsight methods are presented later in Section 4.5. Table 4 shows that the proposed stacked autoencoder as the feature selection model combined with the Improved DeepInsight and GeneViT outperforms other variants of dimensionality reduction techniques for the datasets DS1, DS2, DS3, DS7, DS8, and DS10. Whereas in DS4 and DS5, results are nearly at par with the ReliefF and mRmR methods. For DS6, results are quite close to the CS and ReliefF methods. For DS 9, results of the proposed stacked autoencoder as the feature selection are the best, with the exception of the specificity, in which case, the results are the similar for all the methods.

To determine two critical hyperparameters of a GeneViT architecture, namely number of transformer blocks (layers) and the number of heads in the multi-head attention of GeneViT, experiments were performed with 4 and 8 transformer attention heads, each with 4 and 8 attention heads. For each combination, the experiments is repeated 4 times using all ten datasets with the original and Improved DeepInsight methods. The models' performances are summarized in Table 5. Since GeneViT model having 8 multi-attention heads and 4 transformer blocks combined with the Improved DeepInsight method has shown better performance than other combinations with an average accuracy of more than 95% on all the ten datasets, this combination is selected for the proposed GeneViT model.

4.5. Performance analysis and comparison

The main objective of the current study is to present an end-to-end solution for gene expression based cancer classification over a large set of cancer types and subtypes. The gene expression data is high dimensional and is and highly imbalanced. Therefore data preprocessing techniques play an important role in such a case. Apart from data augmentation to overcome the data imbalance and to increase the number of samples, an stacked autoencoder is used to extract the most useful genes while reducing the data dimension. Then the data is transformed to an image format with the Improved DeepInsight method. Table 6 lists the dimension of a sample for each dataset, its reduced dimension after using the stacked autoencoder, and the transformed output, after the sample passes through the Improved DeepInsight method with channel expansion. Each sample is finally represented as a 3 tuple, i.e., height, width of the image and number of channels. Number of samples are different in each datasets as shown in the Table 2, the height and width of the image in our experiment is constant 64 × 64 and the number of channel varies for each dataset.

The proposed model GeneViT is also compared with nine machine learning, including nature-inspired methods and the state-of-the-art deep learning approaches; the methods selected for the comparison in this work are primarily recent methods employing gene expression data. These methods are as follows: Liu et al. [37], Zeebaree et al. [38], Maniruzzama et al. [12], Shah et al. [40], Kemal Adem [39], Kilicarslan et al. [41], Debata et al. [42], Deng et al. [43] and Dabba Ali et al. [44]. Table 7 shows the quantitative results of the different classification technique on all the ten datasets. The proposed method achieves the highest classification accuracy of 99.20%, 98.50%, 99.21%, 99.37% and 99.7% on 2 class datasets DS1, DS2, DS3, DS4,

Table 4
Ablation study on dimensionality reduction methods, CS: chi square, MI: Mutual Information, ReliefF, mRmR: Minimum Redundancy Maximum Relevance, PCA: Principal component analysis, SAE: Stacked Autoencoder.

Methods	Measures	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10
CS	Accuracy	94.26	92.78	97.39	97.22	95.93	98.61	97.22	91.72	97.63	96.98
	Precision	94.93	93.24	97.50	97.37	95.99	98.67	97.31	89.50	97.92	98.37
	Recall	94.17	92.78	97.41	97.22	95.93	98.61	97.22	91.20	97.78	94.39
	F1 Score	94.23	92.76	97.39	97.22	95.93	98.61	97.22	90.22	97.75	96.19
	Specificity	94.17	92.78	97.41	97.22	95.93	98.31	98.61	93.46	99.20	98.99
MI	Accuracy	96.72	97.42	97.39	98.61	99.13	97.92	98.61	94.48	98.22	98.79
	Precision	96.97	97.55	97.54	98.65	99.14	97.99	98.67	92.45	98.40	98.63
	Recall	96.67	97.42	97.37	98.61	99.13	97.92	98.61	92.45	98.33	99.42
	F1 Score	96.71	97.42	97.39	98.61	99.13	97.90	98.61	92.45	98.31	99.01
	Specificity	96.67	97.42	97.37	98.61	97.13	98.96	98.31	94.04	99.40	99.71
ReliefF	Accuracy	95.90	94.33	96.52	99.31	98.84	98.61	97.92	93.79	98.82	97.28
	Precision	95.90	93.37	95.72	99.32	98.84	98.67	98.04	93.14	98.91	97.25
	Recall	95.91	94.33	96.55	99.31	98.84	98.61	97.92	88.87	98.89	98.71
	F1 Score	95.90	93.73	95.52	99.31	98.84	98.61	97.93	90.44	98.88	97.89
	Specificity	95.91	94.33	96.55	99.31	98.84	98.31	98.96	95.55	99.30	99.33
mRmR	Accuracy	98.36	93.81	93.91	99.31	99.13	97.22	97.92	94.48	97.63	97.28
	Precision	98.04	94.50	94.28	99.32	99.14	97.32	98.04	92.45	97.92	97.25
	Recall	98.33	93.81	93.87	99.31	99.13	97.22	97.92	92.45	97.78	98.71
	F1 Score	98.36	93.79	93.90	99.31	99.13	97.62	97.91	92.45	97.75	97.89
	Specificity	98.33	93.81	93.87	99.31	99.13	98.61	98.96	92.04	99.20	96.30
PCA	Accuracy	96.72	97.42	97.39	98.61	97.13	97.92	98.61	94.48	98.22	98.79
	Precision	96.97	97.55	97.54	98.65	97.14	97.99	98.67	92.45	98.40	98.63
	Recall	96.67	97.42	97.37	98.61	98.13	97.92	98.61	92.45	98.33	99.42
	F1 Score	96.71	97.42	97.39	98.61	97.13	97.90	98.61	92.45	98.31	99.01
	Specificity	96.67	97.42	97.37	98.61	98.13	98.96	98.31	92.04	99.40	99.71
SAE	Accuracy	**99.20**	**98.50**	**99.21**	**99.37**	**99.77**	**98.66**	**99.30**	**95.20**	**99.45**	**99.40**
	Precision	**98.40**	**98.45**	**98.93**	**99.30**	**99.70**	**98.67**	**99.30**	**93.67**	**99.35**	**99.52**
	Recall	**99.50**	**98.37**	**99.45**	**99.30**	**99.70**	**98.61**	**99.30**	**92.87**	**99.40**	**99.55**
	F1-Score	**99.20**	**98.79**	**99.21**	**99.37**	**99.70**	**98.61**	**99.30**	**93.25**	**99.40**	**99.59**
	Specificity	**99.50**	**98.33**	**99.14**	**99.31**	**99.71**	**98.96**	**99.41**	**93.96**	**99.41**	**99.93**

Table 5
Average performance of the proposed method for microarray datasets.

Dataset	4-Head								8-Head						
	Block	Accuracy	Precision	Recall	F1 score	kappa	AUC	Specificity	Accuracy	Precision	Recall	F1 score	kappa	AUC	Specificity
DS1	4	98.40	96.97	98.67	98.40	96.71	98.81	98.33	99.20	98.40	99.50	99.20	98.40	99.59	99.50
	8	96.72	96.97	96.67	96.71	93.43	98.48	96.67	98.36	98.04	98.33	98.36	96.72	99.13	98.33
DS2	4	97.29	97.29	97.29	97.29	95.09	97.95	93.81	98.50	98.45	98.37	98.79	97.20	98.99	98.33
	8	94.33	94.37	94.33	94.33	88.66	97.85	94.33	97.42	97.55	97.42	97.42	94.85	97.79	97.42
DS3	4	96.52	96.75	96.25	96.25	94.08	98.78	93.81	99.21	98.93	99.45	99.21	98.83	99.59	99.14
	8	96.52	96.72	96.55	96.52	93.05	96.04	96.55	97.39	97.54	97.37	97.39	94.78	99.56	97.37
DS4	4	97.91	98.76	97.90	97.90	95.80	97.45	94.75	99.37	99.30	99.30	99.37	98.60	99.31	99.31
	8	93.75	94.44	93.75	93.73	87.50	98.13	93.75	99.13	99.14	99.13	99.13	98.26	99.07	99.13
DS5	4	98.84	98.76	98.95	98.83	97.67	98.40	94.33	99.77	99.70	99.70	99.70	99.84	99.93	99.71
	8	96.22	96.49	96.22	96.22	92.44	98.98	96.22	98.61	98.67	98.61	98.61	97.92	98.82	99.31
DS6	4	96.22	96.54	96.22	96.54	95.83	98.73	96.22	98.66	98.67	98.61	98.61	97.92	99.29	98.96
	8	94.26	94.93	94.17	94.23	88.50	98.84	94.17	97.22	97.44	97.22	97.22	95.83	98.78	98.61
DS7	4	96.61	96.67	96.61	96.61	94.87	99.60	94.96	99.30	99.30	99.30	99.30	99.30	99.78	99.41
	8	95.83	95.83	95.83	95.83	93.75	99.17	97.92	93.92	98.04	97.92	97.91	96.88	98.20	98.96
DS8	4	93.48	91.07	91.29	91.27	91.27	98.60	98.84	95.20	93.67	92.87	93.25	93.13	98.09	93.96
	8	92.41	89.27	90.15	89.68	89.28	96.82	97.42	93.79	93.14	88.87	90.44	91.06	97.53	93.55
DS9	4	98.88	98.86	98.84	98.84	97.67	98.87	96.61	99.45	99.35	99.40	99.40	99.21	99.69	99.41
	8	97.63	97.92	97.78	97.75	96.84	98.67	96.20	98.82	98.91	98.89	98.88	98.42	98.94	98.60
DS10	4	96.91	96.99	96.91	96.89	95.81	97.04	95.96	99.40	99.52	99.55	99.59	98.76	99.90	99.93
	8	97.08	97.15	98.71	97.79	96.30	97.84	98.31	98.79	98.63	99.42	99.01	98.35	99.91	98.71

and DS5, respectively. For multiclass datasets DS5, DS6, DS7, DS8, DS9 and DS10, the proposed method again performs the best with the classification accuracy of 98.66%, 99.30%, 95.20%, 99.45%, and 99.49% respectively. Table 7 shows that the proposed approach, which is based on feature filtering by stacked autoencoder, Improve DeepInsight as the gene expression data conversion into image format, and multi head attention based ViT as the classifier, outperforms existing competitive approaches.

Table 6
Number of selected genes by stacked autoencoder (SAE) and Image dimension after using Improved DeepInsight method, s: samples, h: height, w: width, c: channels.

Dataset	No of Genes	Selected genes by SAE	Image dimension h w c
DS1	2000	1200	(64, 64, 7)
DS2	24481	15910	(64, 64, 55)
DS3	7129	4990	(64, 64, 16)
DS4	7129	5134	(64, 64, 18)
DS5	15154	9100	(64, 64, 41)
DS6	7129	5134	(64, 64, 24)
DS7	12582	7560	(64, 64, 29)
DS8	7129	5134	(64, 64, 12)
DS9	2308	1550	(64, 64, 5)
DS10	12600	7560	(64, 64, 22)

Fig. 5 displays the confusion matrices generated by GeneViT model on 10 distinct gene expression datasets used in this experiment. The misclassification rates for 2-class datasets DS1, DS2, DS3, DS4, and DS5, are 0.0082, 0.0052, 0.0065, 0.0069, and 0.0029, respectively. All the 2-class datasets have low false positive and false negative rates for each class. On the datasets DS6 and DS7 with three classes, the misclassification rates of the proposed GeneViT model are 0.0139 and 0.0139 respectively, while on DS8 and DS9 datasets of four classes, the rates are 0.0348, 0.0237. Further, the misclassification rate is 0.009 for DS10, which belongs to 5 class dataset. Therefore, the method performs reasonably well will low classification errors. Fig. 5 shows that the misclassification rate is somewhat higher in multiclass datasets than that in case of binary class datasets. The ROC curves for all 5 multiclass datasets are presented in Fig. 6. The figures demonstrate that the proposed model has achieved an impressive performance on all the datasets.

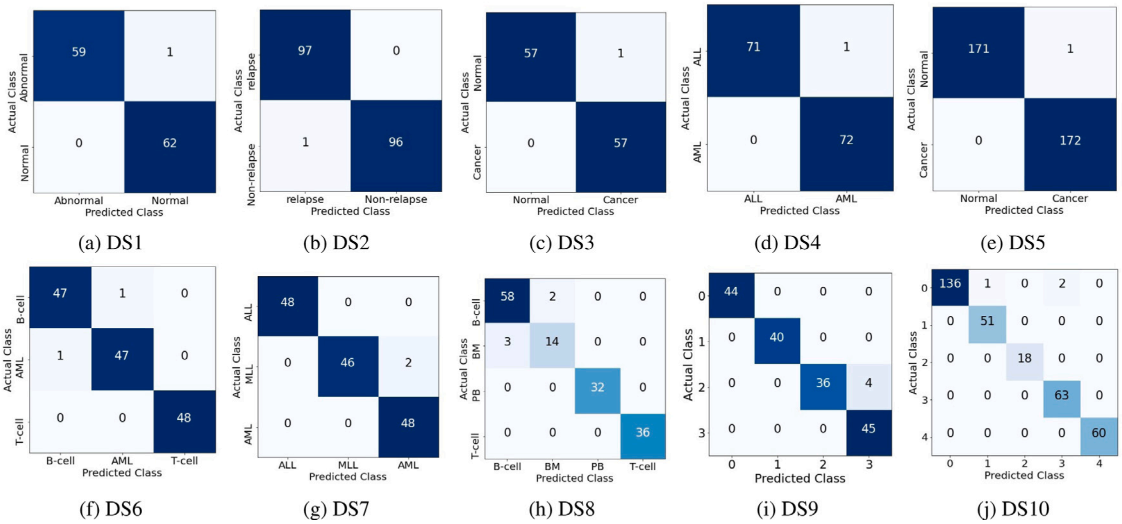
4.6. *Statistical significance*

A statistical test (Student's *t*-test,) is used to check if there is a significant difference in the proposed and other competing methods when it comes to assessing the classification performance. The goal of this test is to verify how effective the proposed method is, in achieving the classification accuracy, as compared with other methods.

We apply a two-tail paired *t*-test on the accuracy values of the all comparing methods. For the sake of brevity, the *t*-test results using only 4 methods, that is 2 machine learning and 2 deep learning methods are shown, namely Maniruzzama et al. [12], Adem [39], Debata et al. [42],

Table 7
Performance comparisons of the proposed model with existing methods.

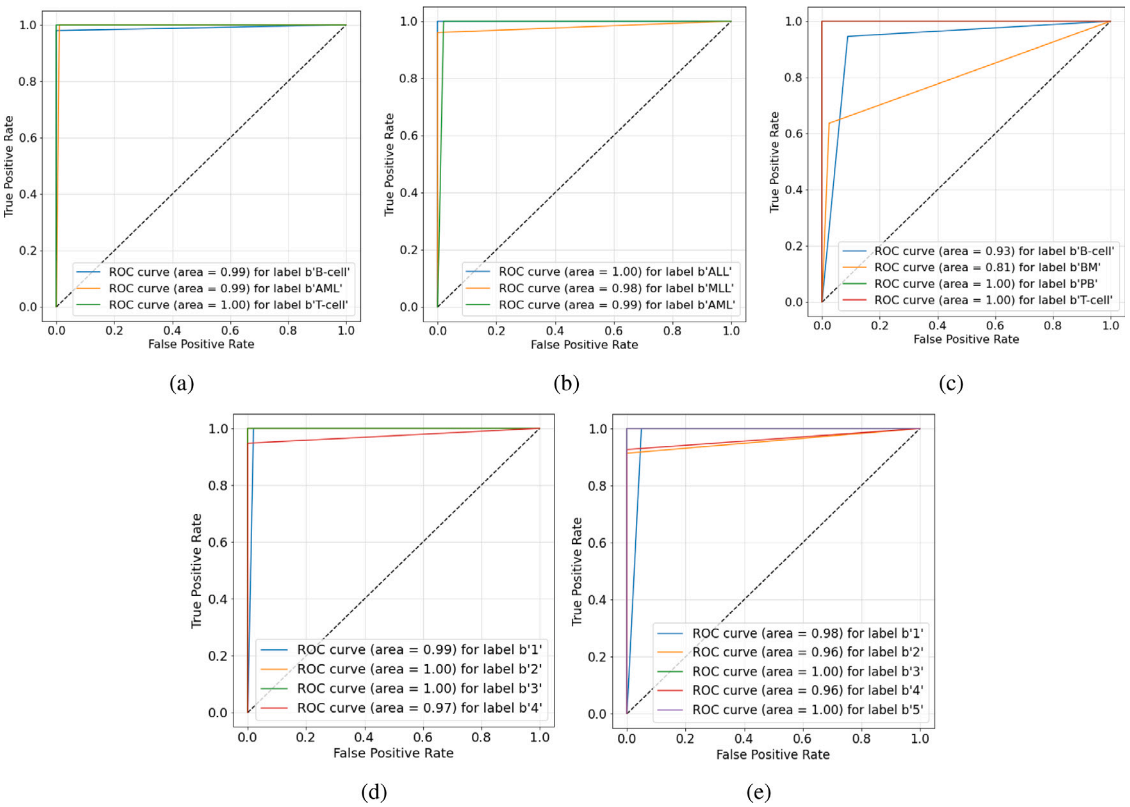
Methods	Measures	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10
Liu et al. [37] 2017	Accuracy	97.57	95.33	57.67	97.67	97.00	98.31	97.29	84.53	96.57	97.98
	Sensitivity	97.83	95.78	64.92	97.05	97.55	98.41	97.55	86.00	96.03	97.63
	Specificity	97.62	95.25	63.02	97.62	97.14	98.93	96.25	82.81	96.15	97.68
	F-1 Score	97.53	95.33	56.12	97.62	95.33	98.33	96.35	83.15	94.81	97.38
Zeebaree et al. [38] 2018	Accuracy	64.50	97.69	98.00	97.00	97.00	98.19	96.57	84.46	94.47	98.19
	Sensitivity	68.81	97.08	98.89	97.94	97.08	97.92	96.41	81.54	94.30	98.17
	Specificity	68.81	98.89	98.75	97.80	96.90	97.62	95.79	82.28	94.30	97.97
	F1-Score	65.00	98.66	98.66	97.57	96.94	97.45	95.78	80.59	94.21	97.95
Maniruzzama et al. [12] 2019	Accuracy	97.63	95.35	96.59	96.33	97.89	98.29	96.29	86.33	96.00	98.39
	Sensitivity	97.83	96.40	91.43	96.67	97.83	98.32	95.55	82.04	95.95	98.56
	Specificity	97.62	95.99	95.63	96.46	97.07	98.67	95.12	82.35	96.91	98.46
	F1-Score	97.53	95.81	96.67	96.39	96.03	98.67	95.01	84.18	95.96	98.47
Shah et al. [40] 2020	Accuracy	**100**	96.00	**100**	99.00	96.89	98.00	97.25	85.25	92.00	97.99
	Sensitivity	**99.89**	96.00	**99.47**	98.62	96.27	98.50	97.07	87.24	93.10	98.14
	Specificity	**99.75**	96.51	**99.49**	98.92	96.44	98.12	97.25	82.75	94.39	98.03
	F1-Score	**98.98**	95.79	**98.89**	98.67	95.72	97.96	96.01	85.29	93.08	97.09
K. Adem [39] 2020	Accuracy	98.23	95.23	93.94	98.33	97.00	98.05	97.07	86.00	95.00	98.44
	Sensitivity	98.91	95.81	94.05	98.30	97.03	98.02	98.44	84.00	96.21	98.49
	Specificity	99.21	95.04	93.75	98.83	97.02	98.09	97.50	86.79	96.37	98.48
	F1-Score	99.82	95.04	93.85	98.93	96.11	98.14	97.70	84.00	96.02	98.32
Kilicarslan et al. [41] 2020	Accuracy	97.56	93.32	83.95	99.06	98.60	97.95	97.33	86.33	97.65	95.56
	Sensitivity	97.43	93.91	81.65	99.07	98.16	97.56	98.00	84.83	96.4	94.03
	Specificity	97.48	93.69	84.99	99.01	98.53	95.05	97.71	83.73	97.46	94.86
	F1-Score	97.19	93.05	85.13	99.05	98.25	97.46	96.84	81.72	96.50	94.23
Debata et al.[42] 2021	Accuracy	98.20	94.3	92.47	99.00	97.58	97.58	98.69	88.10	99.07	98.23
	Sensitivity	98.17	95.98	91.47	98.58	98.38	97.82	98.64	87.00	99.33	98.10
	Specificity	98.53	95.98	91.56	98.33	97.92	98.46	99.00	84.18	99.18	97.39
	F1-Score	98.25	94.70	91.05	98.98	97.94	97.72	98.77	82.82	99.33	97.64
Deng et al. [43] 2022	Accuracy	90.24	82.33	91.67	98.57	**100**	**98.89**	98.44	88.53	98.00	98.89
	Sensitivity	91.17	81.48	92.14	98.89	99.12	97.67	98.50	98.80	87.83	98.67
	Specificity	90.16	81.26	92.16	98.75	99.51	97.75	98.12	98.54	87.07	98.49
	F1-Score	90.18	79.48	91.72	98.66	99.47	97.66	97.96	98.73	88.15	98.89
Dabba Ali et al. [44] 2022	Accuracy	97.10	96.00	96.17	99.10	96.40	98.00	96.30	86.70	**100**	98.10
	Sensitivity	98.60	92.40	97.12	96.70	96.90	98.16	96.80	82.20	**98.45**	98.50
	Specificity	95.80	94.80	96.13	96.70	92.70	98.53	96.20	80.90	**98.47**	98.50
	F1-Score	98.60	95.10	95.19	98.10	93.00	98.25	96.20	84.00	99.11	98.70
Original DeepInsight GeneViT	Accuracy	95.90	93.81	97.39	97.22	98.84	98.02	97.22	91.72	97.63	96.98
	Sensitivity	95.91	93.81	97.41	97.22	97.40	98.19	97.22	91.20	97.78	94.39
	Specificity	95.91	93.81	97.41	97.22	98.84	98.06	98.61	93.46	96.20	97.99
	F1-Score	95.90	93.79	97.39	97.22	98.84	98.18	97.22	90.22	97.75	96.19
Improved DeepInsight GeneViT	Accuracy	99.20	**98.50**	99.21	**99.37**	99.77	98.66	**99.30**	**95.20**	**99.45**	**99.40**
	Sensitivity	99.50	**98.37**	99.45	**99.30**	**99.70**	**98.61**	**99.30**	**92.87**	99.40	**99.55**
	Specificity	99.50	**98.33**	99.14	**99.31**	**99.71**	**98.96**	**99.41**	**93.96**	99.41	**99.93**
	F1-Score	99.20	**98.79**	99.21	**99.37**	**99.70**	**98.61**	**99.30**	**93.25**	**99.40**	**99.59**



(a) DS1 (b) DS2 (c) DS3 (d) DS4 (e) DS5

(f) DS6 (g) DS7 (h) DS8 (i) DS9 (j) DS10

Fig. 5. Confusion matrices generated by the proposed method from 10 datasets, used in this study.



(a) (b) (c)

(d) (e)

Fig. 6. ROC for multi-class datasets, (a) DS6, (b) DS7, (c) DS8, (d) DS9, (e) DS10.

Deng et al. [43], and the proposed method on all the ten datasets. The significance interval is set to 95%, $\alpha = 0.05$. The results of t-test with other methods are also similar to these results. The t-test is based on the Student's t- distribution and is suitable for small samples. In Table 8, a bold p-value indicates that there is no significant difference between the performance of the proposed method and other competing methods. First, all the ten datasets are subjected to a t-test with respect to the accuracy values obtained by the method by Maniruzzama et al. [12] and the GeneViT. The p-values show in the table that the proposed method in [12] have a significant difference in their accuracy results for all the ten datasets. Similarly, the t-test between the method by Adem [39] and the proposed method yields a significant difference on nine datasets, except on DS6 dataset according to the p value which is greater than 0.05 in this case. When we compare the performance of the method by Debata et al. [42] and GeneViT, a p-value greater than 0.05 is observed for only two datasets DS1 and DS4. On all other 8 datasets, the accuracy is shown to be significantly better than the method by Debata et al. [42]. Finally, we perform a t-test between the method by Deng et al. [43] and the proposed method. GeneViT shows significantly better accuracy values on nine datasets except on DS5. We have also conducted t-tests on precision recall, and F1-score, which yield statistically significant findings for $p = 0.01$, implying that the proposed method results are consistent across all the measures.

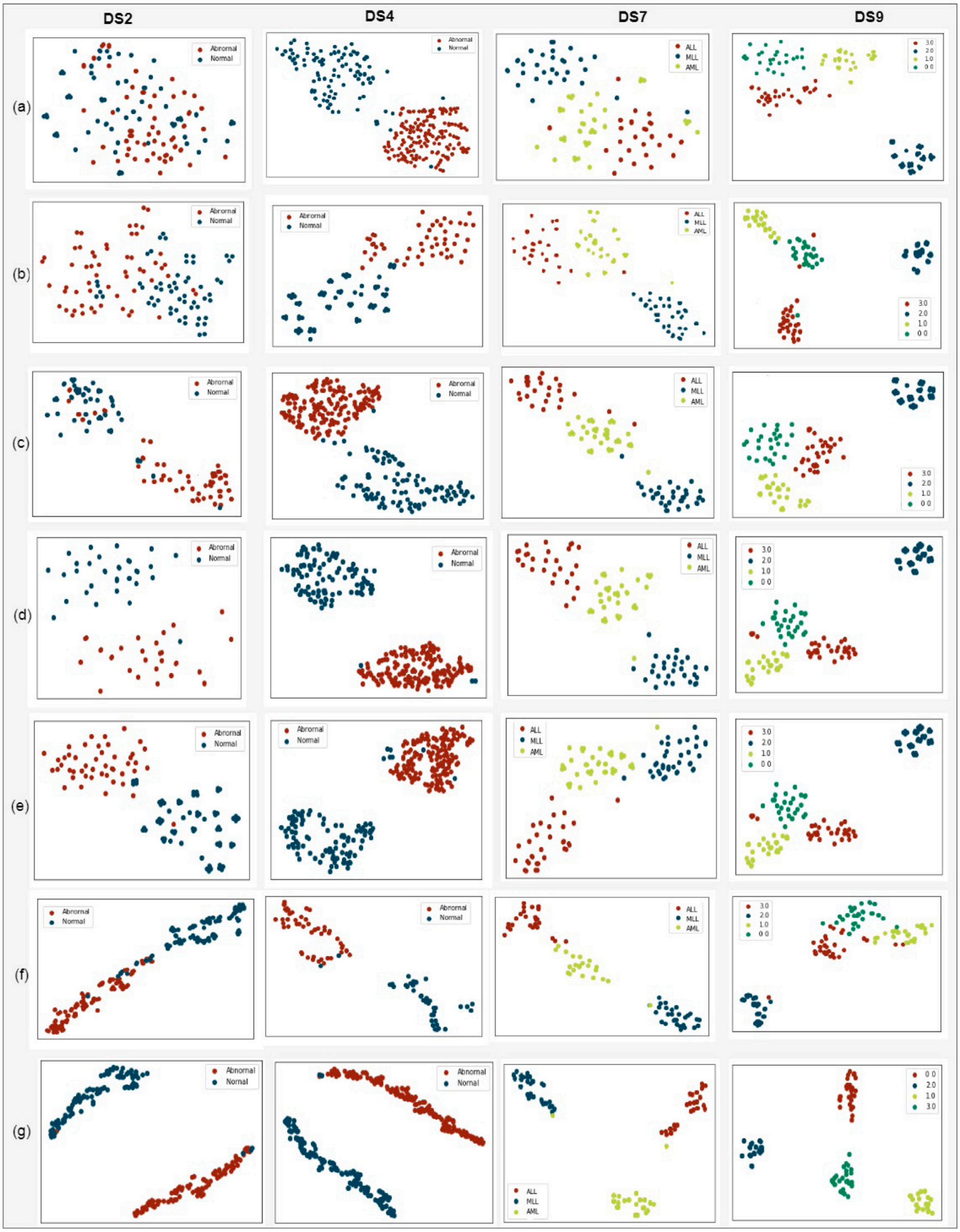


Fig. 7. Comparison of t-SNE plots with other methods (a) Zeebaree et al. (b) Shah et al. (c) Kemal Adem (d) Debata et al. (e) Deng et al. (f) Original DeepInsight (g) Proposed method.

The proposed method is proven to be quite powerful and effective. Additionally, it maintains a fair trade-off between exploration and exploitation of the search area for the issue of gene selection, and provides superior solutions than those of other competing methods. Therefore, one can infer that the performance of GeneViT is statistically significant than other methods.

4.7. Feature set visualization

In order to analyze the effectiveness of the proposed model in extracting right features for cancer data classification, the *t*-SNE method is used to determine the similarity between extracted features of the samples of the same class and dissimilarity for those of samples of different classes in a dataset. Using the *t*-SNE algorithm, the learnt features of various datasets are projected onto the 2D plane [63]. Fig. 7 displays the visualization of 2-D vector scatter plots produced using the *t*-SNE approach for extracted features of samples using GeneViT. The figures present the clusters of projected features for two binary class datasets (DS2-Breast cancer and DS5 Ovarian Cancer), and two multi-class datasets (DS7-MLL and DS9-SRBCT). Additionally, the outcomes are contrasted with the *t*-SNE plots of other deep learning competing

Table 8
Two-tail paired t-test of proposed method and recent methods in the literature.

Datasets	Maniruzzama et al. [12] 2019		K. Adam [39] 2020		Debata et al. [42] 2021		Deng et al. [43]2022	
	t-value	*P*-value	t-value	*P*-value	*t*-value	*P*-value	t-value	*P*-value
DS1	2.891	.018	2.795	.021	−.990	.348	6.295	.000
DS2	3.117	.012	3.180	.011	−97.450	.000	−8.130	.000
DS3	3.586	.006	3.264	.010	−58.395	.000	4.596	.001
DS4	5.779	.000	5.042	.001	−1.960	.082	−8.469	.000
DS5	4.194	.002	3.981	.003	6.433	.000	−.998	.344
DS6	3.770	.004	1.908	.089	3.562	.006	−8.470	.000
DS7	3.095	.013	3.012	.015	1.999	.007	1.746	.015
DS8	4.240	.002	4.318	.003	4.007	.003	3.162	.012
DS9	3.153	.012	2.387	.041	1.837	.049	3.203	.011
DS10	2.795	.021	2.357	.043	2.993	.015	3.113	.012

methods. To compare the unique feature extraction capability of various state-of-the-art approaches, the *t*-SNE features extracted by all of the state-of-the-art methods are compared. The features obtained by the proposed GeneViT are easily distinguishable and form the most appealing clusters among all the methods for both the binary and multiclass datasets as illustrated in Fig. 7(g). Thus, it can be inferred that the proposed approach is extremely effective at comprehending the essential characteristics of various classes in a dataset. Although the results are presented for only four datasets, the same performance is seen on all the datasets.

5. Conclusion and future work

In this study, we propose a novel Gene Vision Transformer model ‘GeneViT’ for cancer classification from the gene expression data. The model contains three main components. The first component deals with data preprocessing and starts with data augmentation to increase the data size and reduce the class imbalance. This is followed by a normalization process of the augmented dataset using the Min–Max statistic approach to ensure a steady convergence of weights and biases. Then, a stacked autoencoder is employed to reduce the dimension of the samples and select the most relevant genes. The second component consists of an Improved DeepInsight method with channel expansion algorithm that converts the data samples into image format to leverage the feature extraction and classification capabilities of the vision transformer. Once the data is converted into images, vision transformer is applied to classify the cancer gene expressions data. The experiments on 10 benchmark datasets demonstrate that the proposed GeneViT with an Improved DeepInsight method outperforms nine state-of-the-art methods in terms of classification accuracy, precision, recall, F1-score, and specificity. Additionally, the method is shown to be quite effective in terms of distinctive feature extraction from data using *t*-SNE plots.

The proposed method could not be evaluated for its cross-dataset performance, as the number of channels produced by the Improved DeepInsight method could be different for different datasets, as a result, GenViT trained on one dataset could not be tested directly on another dataset. In future, the generalizability of the proposed method will be explored using its cross dataset performance, by suitably applying a data conversion approach. Another important aspect in the current scenario is the explainability of the model. A rigorous approach for analyzing the explainability of vision transformer based gene selection and cancer classification model will be explored in future.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The first author acknowledges the research support provided by the Quality Improvement Program of All India Council of Technical Education, Govt. of India.

References

[1] S.A. Raut, S. Sathe, A. Raut, Bioinformatics: Trends in gene expression analysis, in: 2010 International Conference on Bioinformatics and Biomedical Technology, IEEE, 2010, pp. 97–100.
[2] A. Al Kawam, A. Sen, A. Datta, N. Dickey, Understanding the bioinformatics challenges of integrating genomics into healthcare, IEEE J. Biomed. Health Inf. 22 (5) (2017) 1672–1683.
[3] V. Majhi, S. Paul, R. Jain, Bioinformatics for healthcare applications, in: 2019 Amity International Conference on Artificial Intelligence, AICAI, IEEE, 2019, pp. 204–207.
[4] S. Manisekhar, G. Siddesh, S.S. Manvi, Introduction to bioinformatics, in: Statistical Modelling and Machine Learning Principles for Bioinformatics Techniques, Tools, and Applications, Springer, 2020, pp. 3–9.
[5] D. Berrar, M. Granzow, W. Dubitzky, Introduction to genomic and proteomic data analysis, in: Fundamentals of Data Mining in Genomics and Proteomics, Springer, 2007, pp. 1–37.
[6] S. Park, Y. Koh, H. Jeon, H. Kim, Y. Yeo, J. Kang, Enhancing the interpretability of transcription factor binding site prediction using attention mechanism, Sci. Rep. 10 (1) (2020) 1–10.
[7] J. Lee, M. Yoo, J. Choi, Recent advances in spatially resolved transcriptomics: Challenges and opportunities, BMB Rep. 55 (3) (2022) 113.
[8] K. Lan, D.-t. Wang, S. Fong, L.-s. Liu, K.K. Wong, N. Dey, A survey of data mining and deep learning in bioinformatics, J. Med. Syst. 42 (8) (2018) 1–20.
[9] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, J. Saeed, A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction, J. Appl. Sci. Technol. Trends 1 (2) (2020) 56–70.
[10] W. Jia, M. Sun, J. Lian, S. Hou, Feature dimensionality reduction: A review, Complex Intell. Syst. (2022) 1–31.
[11] A.K. Dwivedi, Artificial neural network model for effective cancer classification using microarray gene expression data, Neural Comput. Appl. 29 (12) (2018) 1545–1554.
[12] M. Maniruzzaman, M.J. Rahman, B. Ahammed, M.M. Abedin, H.S. Suri, M. Biswas, A. El-Baz, P. Bangeas, G. Tsoulfas, J.S. Suri, Statistical characterization and classification of colon microarray gene expression data using multiple machine learning paradigms, Comput. Methods Programs Biomed. 176 (2019) 173–193.
[13] E.H. Houssein, D.S. Abdelminaam, H.N. Hassan, M.M. Al-Sayed, E. Nabil, A hybrid barnacles mating optimizer algorithm with support vector machines for gene selection of microarray cancer classification, IEEE Access 9 (2021) 64895–64905.
[14] S. Reddy, A. Kumar, K.Z. Ghafoor, V.P. Bhardwaj, S. Manoharan, CoySvM-(GeD): Coyote optimization-based support vector machine classifier for cancer classification using gene expression data, J. Sensors 2022 (2022).
[15] Y. Wang, X.-G. Yang, Y. Lu, Informative gene selection for microarray classification via adaptive elastic net with conditional mutual information, Appl. Math. Model. 71 (2019) 286–297.
[16] H.H. Aghdam, E.J. Heravi, Guide to Convolutional Neural Networks, Vol. 10, no. 978–973, Springer, New York, NY, 2017, p. 51.
[17] S. Min, B. Lee, S. Yoon, Deep learning in bioinformatics, Brief. Bioinform. 18 (5) (2017) 851–869.
[18] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, J. Dean, A guide to deep learning in healthcare, Nat. Med. 25 (1) (2019) 24–29.

[19] S. Nosratabadi, A. Mosavi, P. Duan, P. Ghamisi, F. Filip, S.S. Band, U. Reuter, J. Gama, A.H. Gandomi, Data science in economics: Comprehensive review of advanced machine learning and deep learning methods, Mathematics 8 (10) (2020) 1799.

[20] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[22] S. Chaudhari, V. Mithal, G. Polatkan, R. Ramanath, An attentive survey of attention models, ACM Trans. Intell. Syst. Technol. 12 (5) (2021) 1–32.

[23] A. Kolesnikov, A. Dosovitskiy, D. Weissenborn, G. Heigold, J. Uszkoreit, L. Beyer, M. Minderer, M. Dehghani, N. Houlsby, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[24] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable detr: Deformable transformers for end-to-end object detection, 2020, arXiv preprint arXiv:2010.04159.

[25] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P.H. Torr, et al., Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 6881–6890.

[26] M. Kumar, D. Weissenborn, N. Kalchbrenner, Colorization transformer, 2021, arXiv preprint arXiv:2102.04432.

[27] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, W. Gao, Pre-trained image processing transformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 12299–12310.

[28] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, C. Schmid, Vivit: A video vision transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6836–6846.

[29] M.M. Naseer, K. Ranasinghe, S.H. Khan, M. Hayat, F. Shahbaz Khan, M.-H. Yang, Intriguing properties of vision transformers, Adv. Neural Inf. Process. Syst. 34 (2021).

[30] E. Portelance, M.C. Frank, D. Jurafsky, A. Sordoni, R. Laroche, The emergence of the shape bias results from communicative efficiency, 2021, arXiv preprint arXiv:2109.06232.

[31] R. Geirhos, K. Narayanappa, B. Mitzkus, T. Thieringer, M. Bethge, F.A. Wichmann, W. Brendel, Partial success in closing the gap between human and machine vision, Adv. Neural Inf. Process. Syst. 34 (2021).

[32] S. Tuli, I. Dasgupta, E. Grant, T.L. Griffiths, Are convolutional neural networks or transformers more like human vision? 2021, arXiv preprint arXiv:2105.07197.

[33] C. Matsoukas, J.F. Haslum, M. Söderberg, K. Smith, Is it time to replace cnns with transformers for medical images? 2021, arXiv preprint arXiv:2108.09038.

[34] A. Sharma, E. Vans, D. Shigemizu, K.A. Boroevich, T. Tsunoda, DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture, Sci. Rep. 9 (1) (2019) 1–7.

[35] E. Alhenawi, R. Al-Sayyed, A. Hudaib, S. Mirjalili, Feature selection methods on gene expression microarray data for cancer classification: A systematic review, Comput. Biol. Med. 140 (2022) 105051.

[36] N. Bhandari, R. Walambe, K. Kotech, S. Khare, Comprehensive survey of computational learning methods for analysis of gene expression data in genomics, 2022, arXiv preprint arXiv:2202.02958.

[37] J. Liu, X. Wang, Y. Cheng, L. Zhang, Tumor gene expression data classification via sample expansion-based deep learning, Oncotarget 8 (65) (2017) 109646.

[38] D.Q. Zeebaree, H. Haron, A.M. Abdulazeez, Gene selection and classification of microarray data using convolutional neural network, in: 2018 International Conference on Advanced Science and Engineering, ICOASE, IEEE, 2018, pp. 145–150.

[39] K. Adem, Diagnosis of breast cancer with stacked autoencoder and subspace kNN, Physica A 551 (2020) 124591.

[40] S.H. Shah, M.J. Iqbal, I. Ahmad, S. Khan, J.J. Rodrigues, Optimized gene selection and classification of cancer from microarray gene expression data using deep learning, Neural Comput. Appl. (2020) 1–12.

[41] S. Kilicarslan, K. Adem, M. Celik, Diagnosis and classification of cancer using hybrid model based on relieff and convolutional neural network, Med. Hypotheses 137 (2020) 109577.

[42] P.P. Debata, P. Mohapatra, A hybrid convolutional neural network approach for feature selection and disease classification, Turk. J. Electr. Eng. Comput. Sci. 29 (SI-1) (2021) 2580–2599.

[43] X. Deng, M. Li, S. Deng, L. Wang, Hybrid gene selection approach using XGBoost and multi-objective genetic algorithm for cancer classification, Med. Biol. Eng. Comput. (2022) 1–19.

[44] A. Dabba, A. Tari, S. Meftali, R. Mokhtari, Gene selection and classification of microarray data method based on mutual information and moth flame algorithm, Expert Syst. Appl. 166 (2021) 114012.

[45] T.M.T. Ab Hamid, R. Sallehuddin, Z.M. Yunos, A. Ali, Ensemble based filter feature selection with harmonize particle swarm optimization and support vector machine for optimal cancer classification, Mach. Learn. Appl. 5 (2021) 100054.

[46] F. Maulidina, Z. Rustam, J. Pandelaki, Lung cancer classification using support vector machine and hybrid particle swarm optimization-genetic algorithm, in: 2021 International Conference on Decision Aid Sciences and Application, DASA, IEEE, 2021, pp. 751–755.

[47] K.D. Sree Devi, P. Karthikeyan, U. Moorthy, K. Deeba, V. Maheshwari, S.M. Allayear, Tumor detection on microarray data using grey wolf optimization with gain information, Math. Probl. Eng. 2022 (2022).

[48] A. Seetharaman, A.C. Sundersingh, Gene selection and classification using correlation feature selection based binary bat algorithm with greedy crossover, Concurr. Comput.: Pract. Exper. 34 (5) (2022) e6718.

[49] M. Lu, Y. Pan, D. Nie, F. Liu, F. Shi, Y. Xia, D. Shen, Smile: Sparse-attention based multiple instance contrastive learning for glioma sub-type classification using pathological images, in: MICCAI Workshop on Computational Pathology, PMLR, 2021, pp. 159–169.

[50] B. Gheflati, H. Rivaz, Vision transformer for classification of breast ultrasound images, 2021, arXiv preprint arXiv:2110.14731.

[51] W. Al-Dhabyani, M. Gomaa, H. Khaled, A. Fahmy, Dataset of breast ultrasound images, Data Brief 28 (2020) 104863.

[52] M.H. Yap, G. Pons, J. Marti, S. Ganau, M. Sentis, R. Zwiggelaar, A.K. Davison, R. Marti, Automated breast ultrasound lesions detection using convolutional neural networks, IEEE J. Biomed. Health Inf. 22 (4) (2017) 1218–1226.

[53] A. Khan, B. Lee, Gene transformer: Transformers for the gene expression-based classification of lung cancer subtypes, 2021, arXiv preprint arXiv:2108.11833.

[54] H. Chen, C. Li, G. Wang, X. Li, M. Rahaman, H. Sun, W. Hu, Y. Li, W. Liu, C. Sun, et al., GasHis-transformer: A multi-scale visual transformer approach for gastric histopathological image detection, Pattern Recognit. (2022) 108827.

[55] Z. Jiang, Z. Dong, L. Wang, W. Jiang, Method for diagnosis of acute lymphoblastic leukemia based on ViT-CNN ensemble model, Comput. Intell. Neurosci. 2021 (2021).

[56] P. Chaudhari, H. Agarwal, V. Bhateja, Data augmentation for cancer classification in oncogenomics: An improved KNN based approach, Evol. Intell. 14 (2) (2021) 489–498.

[57] A. Jain, K. Nandakumar, A. Ross, Score normalization in multimodal biometric systems, Pattern Recognit. 38 (12) (2005) 2270–2285.

[58] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, L. Bottou, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, J. Mach. Learn. Res. 11 (12) (2010).

[59] Y. Bengio, Practical recommendations for gradient-based training of deep architectures, in: Neural Networks: Tricks of the Trade, Springer, 2012, pp. 437–478.

[60] M. Gokhale, S.K. Mohanty, A. Ojha, A stacked autoencoder based gene selection and cancer classification framework, Biomed. Signal Process. Control 78 (2022) 103999.

[61] C. Zhang, Q. Liao, A. Rakhlin, B. Miranda, N. Golowich, T. Poggio, Theory of deep learning IIb: Optimization properties of SGD, 2018, arXiv preprint arXiv:1801.02254.

[62] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (5) (1998) 1299–1319.

[63] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).

[64] G. Sharir, A. Noy, L. Zelnik-Manor, An image is worth 16x16 words, what is a video worth? 2021, arXiv preprint arXiv:2103.13915.

[65] Z. Zhu, Y.-S. Ong, M. Dash, Markov blanket-embedded genetic algorithm for gene selection, Pattern Recognit. 40 (11) (2007) 3236–3248.

[66] S.A. Armstrong, J.E. Staunton, L.B. Silverman, R. Pieters, M.L. den Boer, M.D. Minden, S.E. Sallan, E.S. Lander, T.R. Golub, S.J. Korsmeyer, MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia, Nature Genet. 30 (1) (2002) 41–47.

[67] N.R. Pal, K. Aguan, A. Sharma, S.-i. Amari, Discovering biomarkers from gene expression data for predicting cancer subgroups using neural networks and relational fuzzy clustering, BMC Bioinformatics 8 (1) (2007) 1–18.

[68] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (4) (2009) 427–437.