ArrhythTransform: Multi-Head Attention-Based Transformer Encoder for Arrhythmia Classification

Rahula Shylaja, L. V. Rajani Kumari

Abstract—**Arrhythmia detection based on electrocardiograms (ECG) plays a crucial role in the diagnosis and management of cardiac disorders. Recently, deep learning has shown promising results in healthcare applications. However, the deep neural networks currently employed in the detection of cardiac arrhythmias have excessive complexity, low interpretability, and insufficient learning due to the mechanical stacking of multiple computationally demanding operations. To address these challenges, the proposed work presents a compact transformer encoder model (ArrhythTransform) for the accurate classification of arrhythmia beats from ECG signals. The ArrhythTransform model demonstrates the capabilities of local and global feature extraction by leveraging a multihead attention transformer encoder and a convolutional neural network (CNN) for patch embedding. We have compared the performance of the ArrhythTransform model with a CNN and a multilayer perceptron (MLP) neural network in the embedding block. This work investigates the impact of patch size, model depth, and internal dimensions on overall classification results. Furthermore, the attention distributions from the last transformer layer were presented through attention maps. These attention maps highlight the ECG portions that influence the final predictions, interpreting the decision-making process of the model. For the open-source MIT-BIH arrhythmia database, the proposed ArrhythTransform model with CNN in the embedding block achieved 99.31% accuracy, 98.41% precision, 98.16% recall, and 98.28% F-score in classifying seven unique arrhythmias. The work revealed that careful selection of patch size, patch embedding method, and model hyperparameters improves cardiac arrhythmia classification performance and reduces complexity.**

Index Terms—**Arrhythmia classification, deep learning, electrocardiogram, multi-head attention, Transformer encoder.**

I. Introduction

CARDIAC disorders are still the most significant cause of mortality worldwide [1]. More than 80% of patients with cardiovascular diseases experience arrhythmias, which sometimes lead to severe complications such as stroke and sudden cardiac arrest, which can be life-threatening [2]. Identifying cardiac arrhythmias early may be critical for prompt intervention [3]. The ECG is the most widely used technique for capturing cardiac electrical activity and a common cardiac monitoring tool [4]. However, long-term manual analysis of ECG records is time-consuming, especially in real-time diagnosis. This has generated interest in automated methods to help doctors detect arrhythmias and monitor patients [5], [6].

Deep learning has performed well in numerous applications [7], [8], [9], [10]. The technique has also been used in the classification of ECG signals. CNNs [11], [12], [13], [14], [15], [16], [17], [18] and recurrent neural networks (RNNs) [19], [20], [21], [22] are the architectures most widely used. Özal Yıldırım et al. [3] presented a fast, efficient, and straightforward one-dimensional CNN model to classify 17 types of cardiac arrhythmia disorders. Amin Ullah et al. [23] implemented a two-dimensional CNN for the classification of cardiac arrhythmias. The research transformed one-dimensional electrocardiogram signals into two-dimensional short-time Fourier transform spectrograms. Challenges such as limited training data and moderate computing resources can be resolved using the transfer learning approach [24]. Anita Pal et al. [25] applied the transfer learning technique to the DenseNet architecture, resulting in faster and more accurate heartbeat classification to detect arrhythmias. By fine-tuning and configuring ResNet 50 and AlexNet, Yared Daniel Daydulo et al. [26] achieved the best classification results. Long-term dependencies in the sequences can be tracked and monitored using recurrent neural networks and are widely applied for sequential data processing, such as ECG signals. Somaraju Boda et al. [4] developed a novel architecture using long-term memory (LSTM) for the classification of patient-specific electrocardiogram beats. CNN and RNN architectures have shown effective performance in cardiac arrhythmia classification. However, CNNs struggle to capture long-range temporal dependencies, and RNNs suffer from slow training. Wei Zeng et al. [19] introduced a hybrid model consisting of an LSTM and a one-dimensional CNN architecture for the classification of multiclass arrhythmias. The hybrid architecture of CNN and RNN would capture global and local data, but RNN's sequential process requires much computational power. The transformer architecture has achieved exceptional success in areas such as text classification, segmentation of images, recognition of sound events, and ECG signal processing [27], [28], [29], [30]. The Transformer's self-attention mechanism of the Transformer is dominant in capturing contextual features, while computational efficiency is achieved via its parallel architecture [31]. Duoduo Wang et al. [32] proposed a hybrid electrocardiogram waveform detection model with a combination of a convolutional neural network and a Transformer. Lingxiao Meng et al. [27] proposed the bidirectional Transformer for arrhythmia classification. The architecture improved performance by fully capturing time-dependent factors present in preceding and following contexts. Yong Xia et al. [33] provided a framework that combines CNN with a denoising auto-encoder coupled with a lightweight Transformer, further improving the performance of arrhythmia classification. The multi-head self-attention mechanism proposed by the Google team has demonstrated outstanding performance in machine translation [34]. Yue Wang et al. [35] developed an arrhythmia classification technique using a multi-head self-

Manuscript received June 22, 2025; revised September 19, 2025.

Rahula Shylaja is a Ph.D. scholar (H.T.No: 2301404001) in the Department of Electronics and Communication Engineering, JNT University Hyderabad, Telangana, India (e-mail: shylaja_r@vnrvjiet.in).

L. V. Rajani Kumari is an Associate Professor in the Department of Electronics and Communication Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, affiliated to JNT University Hyderabad, Telangana, India (e-mail: rajanikumari_lv@vnrvjiet.in).

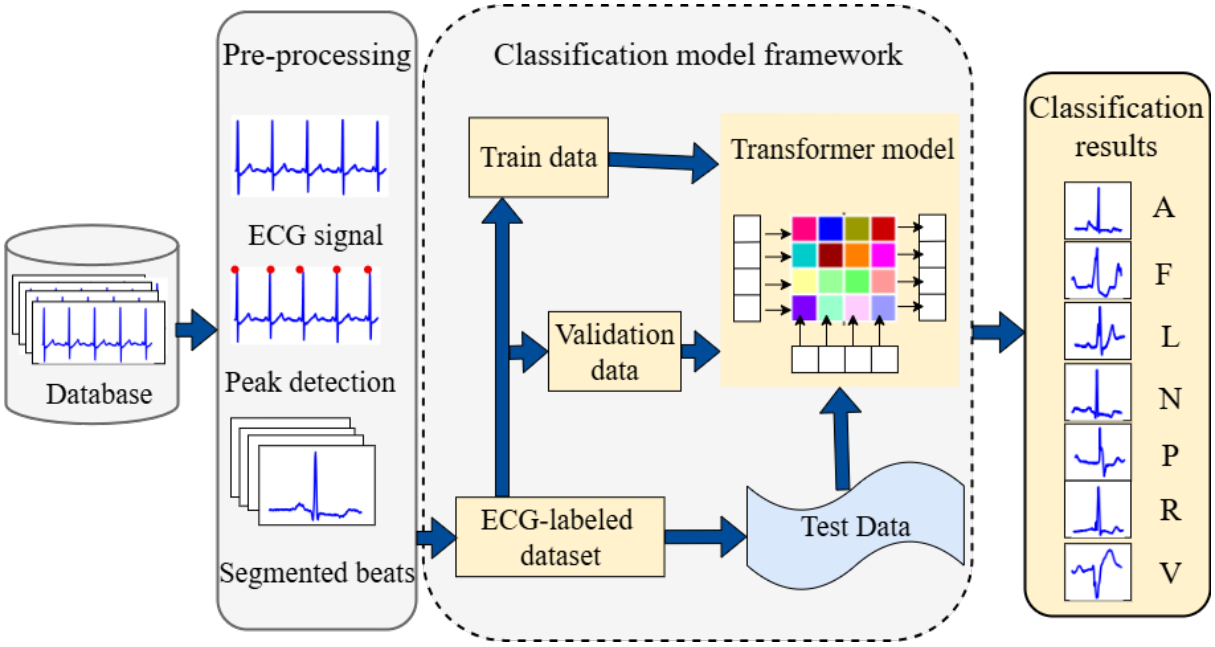

Fig. 1. ECG arrhythmia classification block diagram.

attention mechanism, incorporating positional encoding and linear projection.

Transformer models have particularly excelled in applications that include sequential data analysis. Current transformer-based models have shown promising performance in the classification of cardiac arrhythmia. However, these models employ complicated architecture and are computationally expensive and difficult to interpret. Specifically, transformer architectures are more prone to overfitting, which influences the overall classification accuracy. To address these challenges, we have proposed the ArrhythTransform, a transformer encoder-based model. The ArrhythTransform model enhances performance and reduces complexity by leveraging structured patching and efficient patch embedding. To avoid overfitting during training, learning rate decay and a less complex model with fewer layers were employed.

An overview of the main contributions to the proposed work:

· The transformer encoder-based arrhythmia classification model ArrhythTransform is designed with a multi-head self-attention mechanism.

· The patch embedding and position embeddings are incorporated into the time series ECG signals. The performance of the proposed model is compared with CNN- and MLP-based patch embeddings.

· The proposed architecture's performance is evaluated for various combinations of hyperparameters, including the number of layers, the size of the input patch, and the embedding dimension.

· To understand the decision-making process, the attention scores returned by the last multi-head attention layer of the proposed transformer encoder model are plotted over the ECG heartbeat.

· The proposed method has been compared to state-of-the-art techniques and has produced optimal precision, recall, accuracy, and F-score results.

The remainder of the paper is structured as follows. Section II addresses the methodology, which includes pre-processing of ECG signals, details about the database, the design of a transformer encoder-based classification model, and evaluation metrics. Section III presents the results, while Section IV concludes the proposed work.

TABLE I
ECG BEATS OBTAINED FROM THE MIT-BIH ARRHYTHMIA DATABASE

Class	Train	Validation	Test	Total
N	6952	1005	2043	10000
L	5624	816	1635	8075
R	5075	736	1448	7259
A	1797	247	502	2546
V	5037	690	1403	7130
F	558	89	156	803
P	4946	701	1381	7028
Total beats	29989	4284	8568	42841

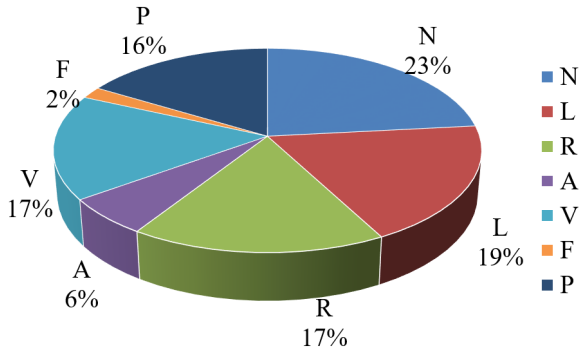

Fig. 2. ECG beat class distribution.

II. Methodology

The proposed ArrhythTransform model investigates the application of the transformer encoder architecture for the classification of ECG arrhythmias. Figure 1 shows the block diagram. The approach is divided into two processing phases:

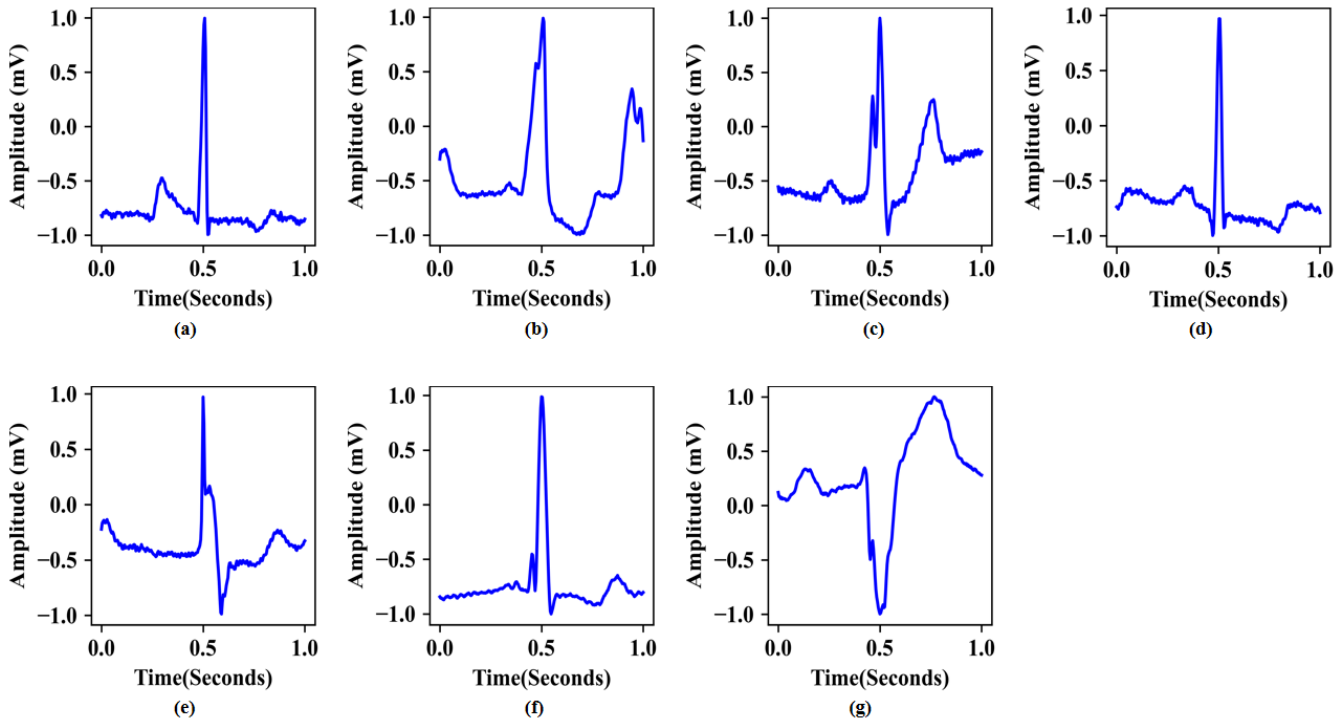

Fig. 3. Illustration of ECG beats from the classes (a) A, (b) F, (c) L, (d) N, (e) P, (f) R, and (g) V.

preprocessing electrocardiogram records taken from the MIT-BIH database and transformer-based arrhythmia classification.

A. Database

The ECG signals used to develop the proposed model are gathered from the MIT-BIH arrhythmia database [36], [37]. The database contains 48 two-channel ambulatory records of 30-minute duration. The recordings were digitized at a 360 Hz sampling rate with a resolution of 11 bits over a 10 mV range. This database provides 17 types of heartbeats with respective annotations.

B. Preprocessing

The collected ECG signals are initially down-sampled to 256 Hz to reduce computational complexity while preserving essential information. Heartbeat segments of 256 samples are extracted from each record using the provided R-peak annotations. Arrhythmia classification commonly follows the Association for the Advancement of Medical Instrumentation (AAMI) standard. This work considers the fusion beat (F), atrial premature contraction (A), normal beat (N), left bundle branch block (L), paced beat (P), right bundle branch block (R), and premature ventricular contraction (V) arrhythmia beats from the MIT-BIH database. Table I displays the selected classes and the respective sample count. Figure 2 shows the class distribution of ECG beats, and Figure 3 shows the types of ECG arrhythmia beats selected from the database.

C. Classification

Figure 4 presents a multi-head self-attention-based transformer encoder architecture. The model incorporates an embedding block, the stacked transformer encoder layers, and an MLP head. The embedding block includes a patch and a position embedding. The transformer encoder includes layer normalization, residual connections, and multiple layers of multi-head attention. Each heartbeat segment is split into a sequence of patches of identical size. Each patch is projected into a high-dimensional space, resulting in a patch embedding vector, which is fed as input to the model. A one-dimensional (1D) CNN or an MLP layer is commonly used for patch embedding. To facilitate the learning of spatial relationships, positional encodings $PosE$ are included as follows:

$$\mathrm{PosE}_{(pos,i)} = \sin\left(\frac{pos}{10000^{\frac{i}{d_k}}}\right) \tag{1}$$

$$\mathrm{PosE}_{(pos,i)} = \cos\left(\frac{pos}{10000^{\frac{i}{d_k}}}\right) \tag{2}$$

where Pos indicates the patch position, i is the dimension index, and d_k is the dimensionality of the transformer encoder model.

The self-attention procedure determines the attention score (Atn) between each element of the sequence. As depicted in Figure 5, self-attention is performed by performing scaled dot product attention. Attention is calculated as follows:

$$\mathrm{Atn}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \tag{3}$$

where Q is Query, V is Value, and K is Key. These metrics are derived from the sequence given as input. d_k represents the dimension of the K vector and is used for scaling to maintain stability in gradients. The Softmax function normalizes the scores, ensuring that they sum up to 1.

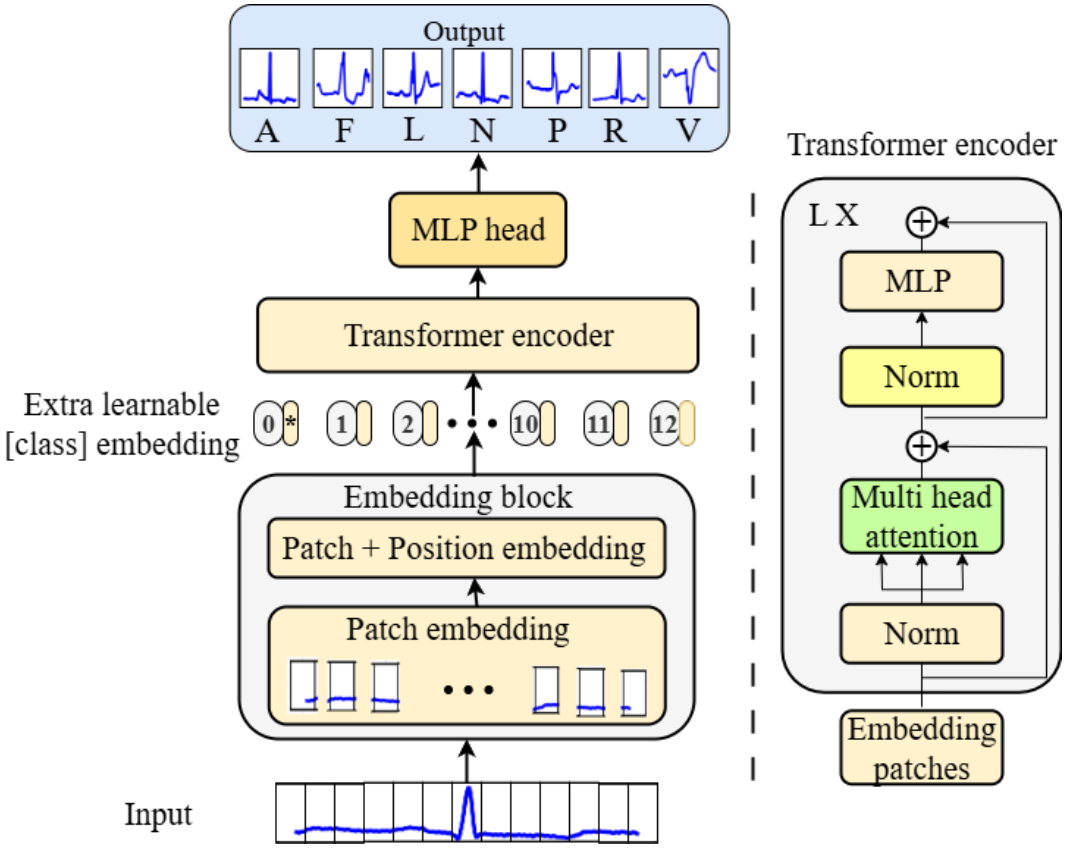

Fig. 4. Proposed transformer model architecture for arrhythmia classification.

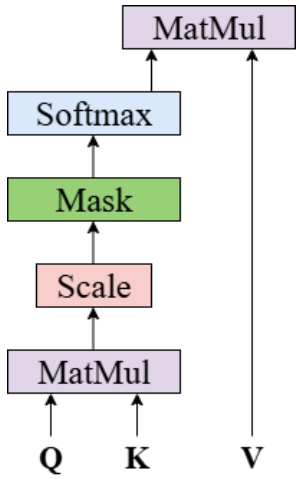

Fig. 5. Self-attention.

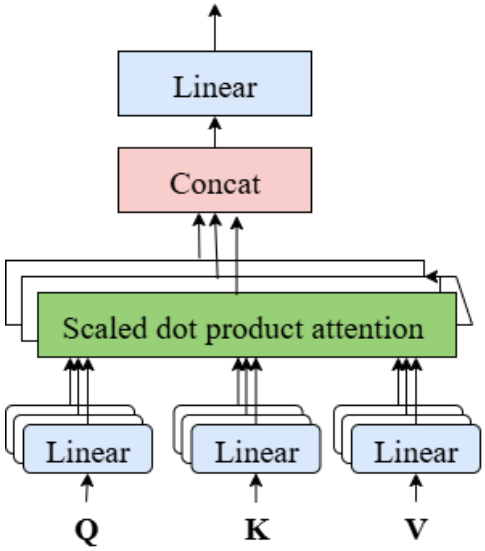

Fig. 6. Multi-head attention.

Transformers use multiple attention heads to process information simultaneously. Figure 6 shows the multi-head attention module. This module uses multiple scaled dot product units. Each head focuses on distinct input data features to capture diverse contextual relationships.

$$\text{MultiHeadAttention}(Q, K, V) = \text{Concat}(\text{h}_1, \ldots, \text{h}_h)W^O \tag{4}$$

$$\text{h}_i = \text{Attn}(QW_i^Q, KW_i^K, VW_i^V) \tag{5}$$

Where W_i^Q, W_i^K, and W_i^V represent each attention head's (h_i) learnable weight matrices. The concatenated outputs are projected back to the desired dimension using the output weight matrix W^O.

Each layer in the transformer encoder includes a feed-forward neural network with a GELU activation function, residual connections, and normalization layers. These components help the model learn complex patterns efficiently while stabilizing the training. The model uses a special learnable classification (CLS) token, a comprehensive representation of the entire sequence. The CLS token interacts with other patch embeddings through self-attention, and its output is used for classification tasks. Tables II and Table III show the model parameter specifications.

TABLE II
THE MODEL PARAMETER SPECIFICATIONS

Parameter	Value
Number of heads	4
Patch embedding size	64
Patch size	32
MLP Unit	[64, 32]
Encoder layers	2
Input Dimension	1×256
Output Dimension	7 classes
Batch size	32
Epochs	100
Learning rate	0.0001
Loss function	Categorical cross-entropy
Optimizer	Adam
Activation functions	Gelu, Softmax
Parameters	44935

TABLE III
PATCH EMBEDDING AND TRANSFORMER ENCODER HYPERPARAMETER DETAILS

Patch Embedding	
X (Input)	[1, 256]
Convolution Layer	Kernel size: 1×32 Stride: 32 Number of filters: 64 in: 1×256 out: 8×64
Transformer Encoder	
X (input)	[9, 64]
Multi-head attention	Number of layers: 2 W^Q, W^K, W^V, W^O: 64×64
Self-attention	In: $X \cdot W^Q : [9 \times 64] \Rightarrow Q : 9 \times 64$ $X \cdot W^K : [9 \times 64] \Rightarrow K : 9 \times 64$ $X \cdot W^V : [9 \times 64] \Rightarrow V : 9 \times 64$ Out: 9×64
Layer Norm	[9, 64]
MLP	In: 64, Out: 32
Layer norm	[9, 64]
Softmax	[64, 7]

Algorithm 1 describes the sequential steps in the proposed architecture of the model. The input consists of ECG records retrieved from the database. Each record contains a sequence of heartbeats with annotated labels. The output is one of the seven types of arrhythmias that are considered for classification. Step 1 is data preprocessing, which involves segmentation based on R-peak annotations, down-sampling, and vector representation of the ECG signal input. Step 2 provides details abouta set splitting. Step 3 shows parameter details such as signal length, batch size, and number of channels. The number of channels is one, as ECG is a 1-dimensional time series signal.

Step 4.1 patches the sequence, provides patch embeddings using 1D CNN, adds position encoding to the patch embedding, and concatenates the trainable class token to prepare the input for the transformer encoder. In Step 4.2.1, training occurs on multiple layers that contain transformer encoders consisting of multi-head attention, followed by layer normalization and an MLP layer. The last layer output is processed by a dense layer with softmax activation, which generates class probabilities in step 4.2.2. Step 5 returns the final class label of the input.

The proposed transformer model adapts the Adaptive Moment Estimation (Adam) optimizer. Adam brings gradient descent together with the momentum algorithm and the RMSProp algorithm, resulting in a more efficient and robust process for training neural networks. Adam adjusts the learning rate for each parameter dynamically based on the moments of the gradients and incorporates momentum to smooth out the optimization process. For a parameter at timestamp t, the moment update equations take the following form:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{6}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{7}$$

where g_t denotes the gradient at time step t and β_1, β_2 are the decay rates for the two moments. The formulas for bias correction and updating parameters are given as follows:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{8}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{9}$$

$$\omega_{t+1} = \omega_t - \frac{\eta}{\sqrt{\hat{v}_t} + \varepsilon} \hat{m}_t \tag{10}$$

where η denotes the learning rate and ε represents a positive constant with a small value to avoid division by zero.

The proposed model utilized a categorical cross-entropy loss function during training to adjust the model weights, resulting in a minimum loss. Categorical cross-entropy loss, known as softmax loss, is normally used in multiclass classification. With softmax activation applied to the neural network's raw outputs, the result of multiclass classification is a vector of probabilities predicted over the different classes of input heartbeats. The categorical cross-entropy loss for a single sample is given as follows:

$$\text{Loss} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{11}$$

where y_i represents the target, $\hat{y}_i$ is the estimated output, and C denotes the number of classes.

D. Performance evaluation metrics

The performance of the proposed architecture is measured using the metrics recall, precision, F-score, and accuracy. Accuracy measures the correctness of the classification model. For accurate positive correctness, the F-score combines recall and precision. The metrics are computed using a confusion matrix with a true class present in each row and the predicted class in each column, as illustrated in Table IV.

From the confusion matrix, the following formulae can be derived.

$$\text{Overall Accuracy} = \frac{\sum_{i=1}^{N} \text{Cl}_{ii}}{\sum_{i=1}^{N} \sum_{j=1}^{N} \text{Cl}_{ij}} \tag{12}$$

Algorithm 1: The algorithm of the proposed transformer encoder-based arrhythmia classification model

Input	ECG record $X(t) = \{x_1, x_2, \ldots, x_n\}$
	$n \rightarrow$ The total count of samples in the ECG record
	$x_t \rightarrow$ Amplitude of the signal at time t
Output	Classification result of ECG Arrhythmia class = {A, F, L, N, P, R, V}
Step 1:	Preprocessing
Step 1.1:	Segment heartbeats using R-peak annotations.
	$R = \{r_1, r_2, r_3, \ldots, r_m\}$
	$m \rightarrow$ The number of R-peaks in a record
	$r_j \rightarrow$ R-peak of j^{th} heartbeat in the record
	For each r_j with class label L, heartbeats are segmented using a window of 360 samples centered around r_j
	$H_j = X[r_j - 180 : r_j + 180]$
Step 1.2:	Down-sample each heartbeat with label L to 256 Hz
	$S_j = H_j[:: 360/256] \in \mathrm{R}^{256}$
	Training data sample with label L: $S_j^L = \{S_1^L, S_2^L, \ldots, S_{256}^L\}$
Step 2:	Data set splitting
	Train $\rightarrow$ 70%, Validation $\rightarrow$ 10%, Test $\rightarrow$ 20%
Step 3:	Input ECG training vector representation
Step 3.1:	Input shape = (signal length, number of channels)
	Input tensor $S \in \mathrm{R}^{B \times L \times C}$
	$B \rightarrow$ batch size, $L \rightarrow$ signal length, $C \rightarrow$ number of channels
Step 4:	For each batch
Step 4.1:	Prepare transformer encoder input
Step 4.1.1:	Use CNN to make patches.
	$P = \text{Conv1D}(S, F, K, S_r)$
	$F \rightarrow$ number of filters, $K \rightarrow$ kernel size, $S_r \rightarrow$ stride length
	where $P \in \mathrm{R}^{B \times N \times M}$ and $N = \frac{L-K}{S_r} + 1$ is the number of patches.
Step 4.1.2:	Positional embedding
	$E_{\text{pos\_token}} = \text{PosEmbedding}(N, F)$ is assigned.
	The output expands to $E_{\text{pos\_token}} \in \mathrm{R}^{B \times N \times M}$
Step 4.1.3:	Add patch and position embeddings:
	$E = P + E_{\text{pos\_token}}$
Step 4.1.4:	Concatenate class token.
	cls $\in \mathrm{R}^{B \times 1 \times M}$ and $X_{\text{out}} \in \mathrm{R}^{B \times (N+1) \times M}$
Step 4.2:	For i = 0 to B
Step 4.2.1:	For each transformer encoder layer,
Step 4.2.1.1:	Initialize query, key, and value matrices.
	$Q = X_{\text{out}} \cdot W^Q$, $K = X_{\text{out}} \cdot W^K$, $V = X_{\text{out}} \cdot W^V$
Step 4.2.1.2:	Multi-head attention matrix is evaluated.
	$X_{\text{out}} = \text{MultiHeadAttention}(Q, K, V)$
Step 4.2.1.3:	*LayerNormalization*(X_{out})
Step 4.2.1.4:	$X_{\text{out}} = \text{MLP}(X_{\text{out}})$
Step 4.2.1.5:	*LayerNormalization*(X_{out})
Step 4.2.3:	Append class
Step 4.2.4:	$P = \text{softmax}(\text{cls token})$
Step 5:	Return predicted class

TABLE IV
CLASS N CONFUSION MATRIX

	Predicted class			
True class	Class 1	Class 2	....	Class N
Class 1	Cl_{11}	Cl_{12}	....	Cl_{1N}
Class 2	Cl_{21}	Cl_{22}	....	Cl_{2N}
...	...	...	....	...
Class N	Cl_{N1}	Cl_{N2}	....	Cl_{NN}

$$\text{Recall(Class } i) = \frac{\text{Cl}_{ii}}{\sum_{j=1}^{N} \text{Cl}_{ij}} \tag{13}$$

$$\text{Precision(Class } i) = \frac{\text{Cl}_{ii}}{\sum_{j=1}^{N} \text{Cl}_{ji}} \tag{14}$$

$$F\text{-score} = \frac{2 \cdot \text{Precision(Class } i) \cdot \text{Recall(Class } i)}{\text{Precision(Class } i) + \text{Recall(Class } i)} \tag{15}$$

III. RESULTS

The experiments were conducted using Python on a system powered by a 13th Generation Intel(R) Core(TM) i7-13700 CPU @ 2.1 GHz with thirty-two gigabytes of RAM and an NVIDIA GeForce RTX 4080. The model was developed using the parameters shown in Table II and Table III. Table III shows the hyperparameter specifications of each layer. The Adam optimizer was used for model training for

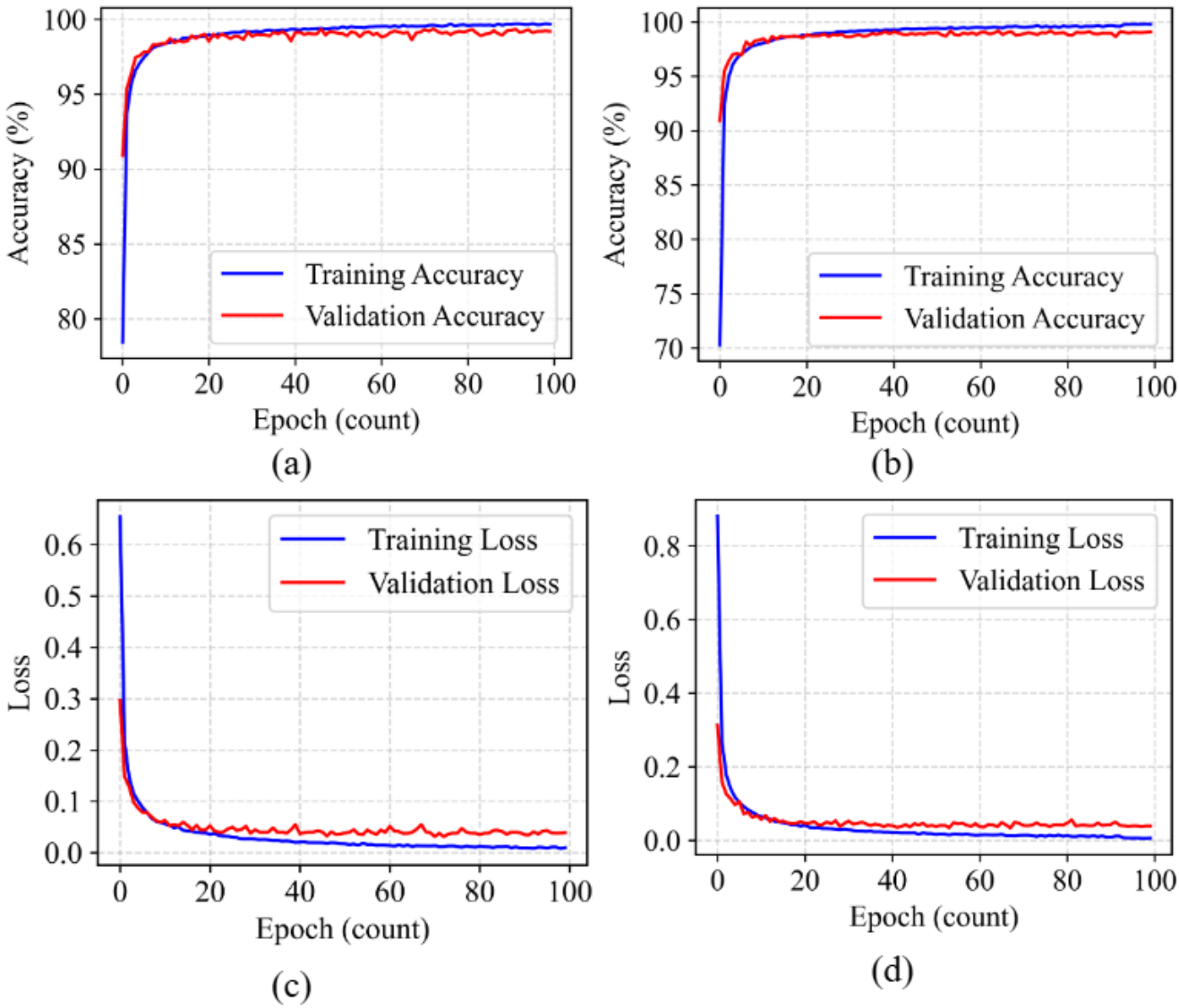

Fig. 7. Model performance: (a) Accuracy plot of the model with CNN in embedding block, (b) Accuracy plot of the model with MLP in embedding block, (c) Loss plot of the model with CNN in embedding block, and (d) Loss plot of the model with MLP in embedding block.

TABLE V
PERFORMANCE MEASURES OF THE MODEL WITH CNN AND MLP FOR PATCH EMBEDDING

Class	CNN for Patch embedding			MLP for Patch embedding			
	Precision (%)	Recall (%)	F-score (%)	Precision (%)	Recall (%)	F-score (%)	Support
A	97.25	98.61	97.92	97.82	98.41	98.11	502
F	94.04	91.03	92.51	90.51	91.67	91.08	156
L	99.69	99.57	99.63	99.45	99.63	99.54	1635
N	99.76	99.76	99.76	99.66	99.66	99.66	2043
P	99.93	99.93	99.93	99.78	99.78	99.86	1381
R	99.72	99.79	99.76	99.45	99.72	99.59	1448
V	98.50	98.43	98.47	98.42	97.72	98.07	1403
Macro avg	98.41	98.16	98.28	97.89	98.08	97.99	–
Weighted avg	99.31	99.31	99.31	99.15	99.15	99.15	–
Accuracy	**99.31**			**99.15**			–

100 epochs at a learning rate of 0.0001. A batch size of thirty-two was used. The categorical cross-entropy loss function was used for the experiment. Instead of using pre-trained weights, the model in this work was trained from scratch using the architectural design.

We evaluated the effectiveness of CNN and MLP for patch embedding. In addition, we experimented with the Arrhyth-Transform model using several hyperparameters, including the head count (4, 8), the number of layers (2, 4), and the patch embedding dimension (32, 64) for varying patch sizes (8, 16, and 32). Learning rate decay and a lower-complexity model with 2 and 4 layers are employed to mitigate the risk

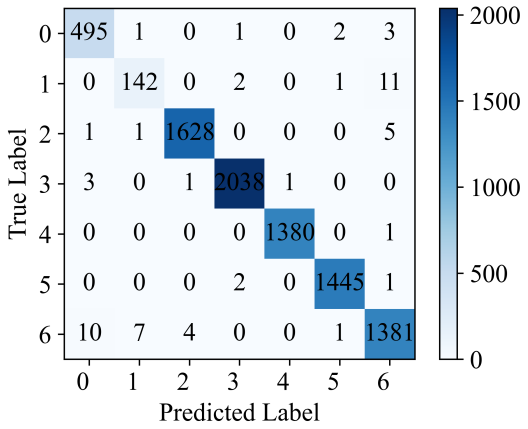

Fig. 8. Confusion matrix of the model with CNN in the embedding block.

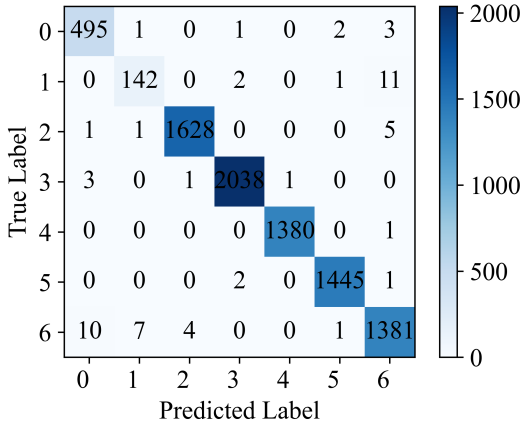

Fig. 9. Confusion matrix of the model with MLP in the embedding block.

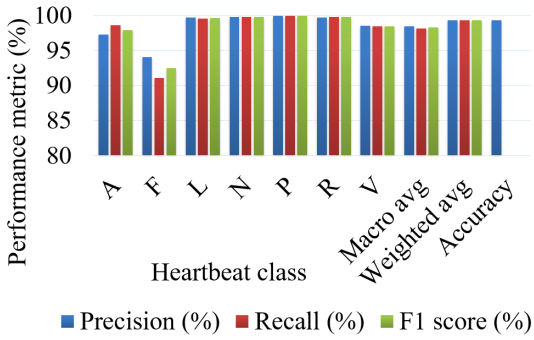

Fig. 10. Performance metrics of the model using CNN for patch embedding

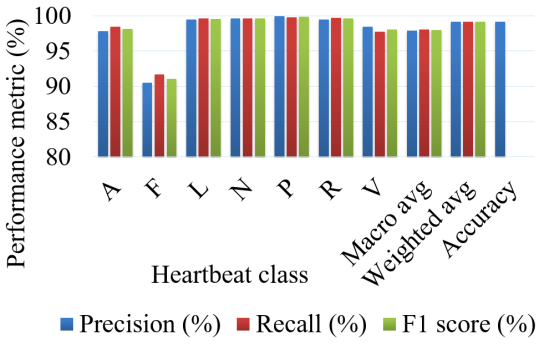

Fig. 11. Performance metrics of the model using CNN for patch embedding

TABLE VI
COMPARISON OF PERFORMANCE METRICS FOR CNN- AND MLP-BASED PATCH EMBEDDING

Performance metric	Patch Embedding with CNN	Patch Embedding with MLP
Accuracy (%)	99.31	99.15
Precision (%)	98.41	97.89
Recall (%)	98.16	98.08
F-score (%)	98.28	97.99

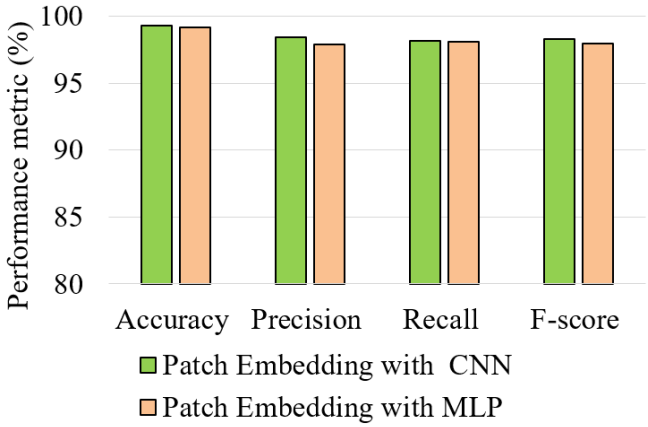

Fig. 12. Comparison of performance metrics for CNN- and MLP-based patch embedding

of overfitting during training.

To avoid losing shape information, the heartbeat-segmented patches were mapped to a high-dimensional space. Model performance is affected by the patch embedding technique in the embedding block. We evaluated the performance of the ArrhythTransform model using CNN and a multi-layer perceptron (MLP) in the embedding block. The validation, training accuracy, and loss plots of the model using CNN for patch embedding are shown in Figure 7(a) and 7(c). The validation, training accuracy, and loss plots of the model using MLP for patch embedding are shown in Figure 7(b) and 7(d). These plots show the performance improvement of the models over time.

The confusion matrices of the model with CNN and MLP in the embedding block are presented in Figures 8 and 9. Performance metrics, including precision, accuracy, recall, and F-score, can be assessed using the confusion matrix. Figure 10, Figure 11, and Table V present the performance of the model using CNN and MLP in the embedding block for various classes of cardiac arrhythmia beats. The macro average (Avg) and the weighted average of precision, recall, F-score, and overall classification accuracy are also presented in the table.

The proposed model achieved an average accuracy of 99.31% using CNN for patch embedding. The model achieved the highest precision of 99.93% for class P, and class F had the lowest precision of 94.92%, indicating that 99.93% of class P instances were correctly classified and 94.04% of class F instances were correctly predicted. The model achieved 99.93% recall for class P, while class F achieved the lowest recall of 91.03%, indicating that 99.93% class R instances were identified, while only 91.03% of the actual class F instances were correctly classified. The F-

TABLE VII
PERFORMANCE COMPARISON OF THE PROPOSED MODEL TRAINED WITH DIFFERENT HYPERPARAMETER SETTINGS, REPRESENTED AS AT-I TO AT-VI

Model	L	ED	H	Precision (%)		Recall (%)		F-score (%)		Accuracy (%)	TP
				Macro avg	Weighted avg	Macro avg	Weighted avg	Macro avg	Weighted avg		
AT I	2	32	4	98.35	99.24	97.78	99.24	98.05	99.24	99.24	18471
AT II	2	32	8	98.19	99.20	97.73	99.21	97.95	99.20	99.20	18471
AT III	4	32	4	97.76	99.14	97.96	99.14	97.86	99.14	99.14	35559
AT IV	4	32	8	97.77	99.05	97.29	99.05	97.53	99.05	99.05	35559
AT V	2	64	4	**98.41**	**99.31**	**98.16**	**99.31**	**98.28**	**99.31**	**99.31**	44935
AT VI	2	64	8	98.38	99.29	98.16	99.29	98.27	99.29	99.29	44935

AT: ArrhythTransform, L: Number of layers, ED: Embedding dimension, H: Number of heads, TP: Trainable parameters.

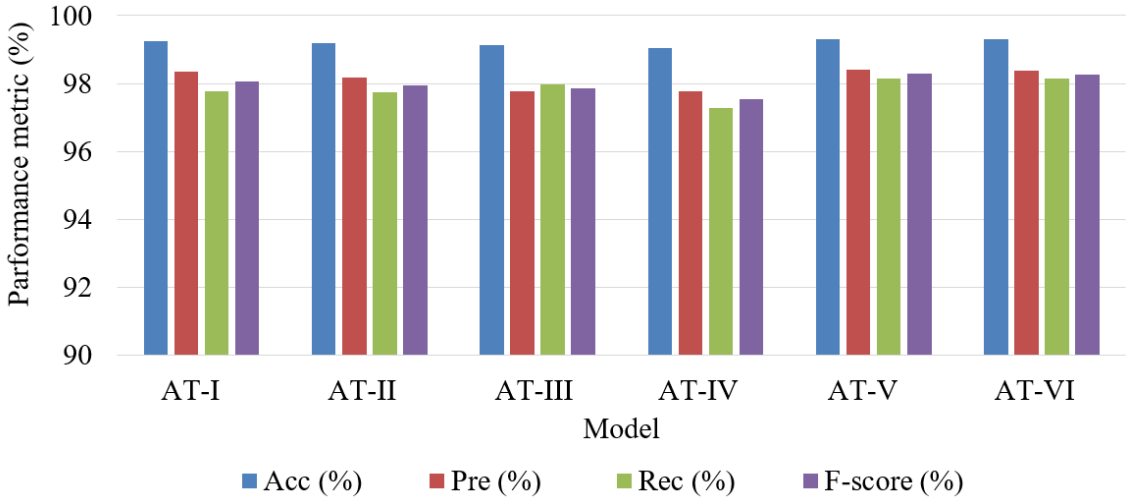

Fig. 13. Performance comparison of the proposed model trained with different hyperparameter settings, represented as AT-I to AT-VI

TABLE VIII
INDIVIDUAL CLASS PERFORMANCE OF THE PROPOSED MODEL TRAINED WITH DIFFERENT HYPERPARAMETER SETTINGS, REPRESENTED AS AT-I TO AT-VI

Performance metric	Class	Model					
		AT-I	AT-II	AT-III	AT-IV	AT-V	AT-VI
Precision (%)	A	97.44	97.82	98.20	97.42	97.25	98.02
	F	93.88	92.62	89.31	90.60	94.04	93.42
	L	99.51	99.45	99.69	99.63	99.69	99.63
	N	99.85	99.85	99.61	99.66	99.76	99.80
	P	99.86	99.93	99.71	99.93	99.93	99.86
	R	99.59	99.59	99.65	99.52	99.72	99.66
	V	98.29	98.08	98.14	97.67	98.50	98.30
Recall (%)	A	98.61	98.41	98.01	97.61	98.61	98.61
	F	88.46	88.46	91.03	86.54	91.03	91.03
	L	99.57	99.63	99.57	99.27	99.57	99.63
	N	99.61	99.61	99.66	99.56	99.76	99.46
	P	99.93	99.86	99.78	99.86	99.93	99.86
	R	99.79	99.79	99.65	99.79	99.79	99.79
	V	98.50	98.36	98.00	98.43	98.43	98.72
F-score (%)	A	98.02	98.11	98.11	97.51	97.92	98.31
	F	91.09	90.49	90.16	88.52	92.51	92.21
	L	99.54	99.54	99.63	99.45	99.63	99.63
	N	99.73	99.73	99.63	99.61	99.76	99.63
	P	99.89	99.89	99.75	99.89	99.93	99.86
	R	99.69	99.69	99.65	99.66	99.76	99.72
	V	98.40	98.22	98.07	98.05	98.47	98.51

score is the harmonic mean of precision and recall. The F-score considers false positives and false negatives, offering a balanced measure between the two metrics. The model achieved the highest balanced performance for category P,

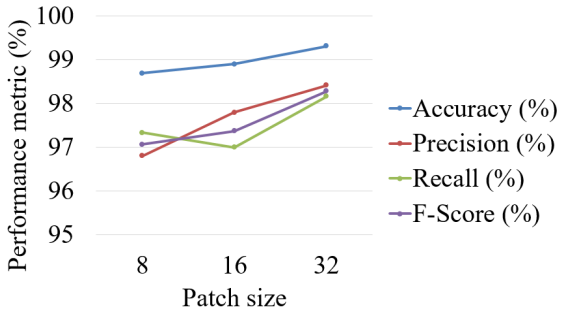

Fig. 14. Comparison of statistical results of the models with different patch dimensions for a patch embedding size of 64 with 4 heads

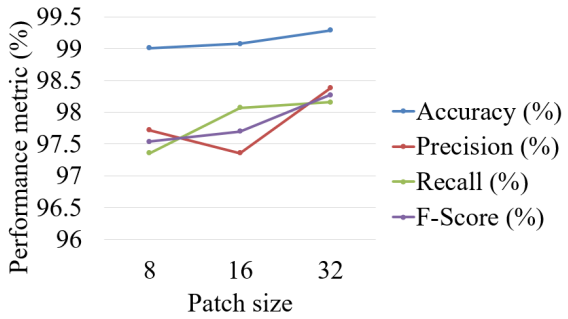

Fig. 15. Comparison of statistical results of the models with different patch dimensions for a patch embedding size of 64 with 8 heads

TABLE IX
PERFORMANCE COMPARISON WITH DIFFERENT PATCH SIZES AND NUMBER OF HEADS

	Model with 4 heads			Model with 8 heads		
Patch Size	8	16	32	8	16	32
Accuracy (%)	98.69	98.90	99.31	99.01	99.08	99.29
Precision (%)	96.80	97.79	98.41	97.72	97.36	98.38
Recall (%)	97.33	96.99	98.16	97.36	98.07	98.16
F-score (%)	97.06	97.37	98.28	97.54	97.70	98.27

producing an F-score of 99.93% and the least balanced performance for class F with an F-score of 92.51%.

The relative performance comparison between the model with CNN and MLP in the embedding block for patch embedding is presented in Table VI and Figure 12. Figure 12 illustrates macro-averaged precision, recall and F-score values. The model with CNN for patch embedding achieved better performance, as the transformer encoder's multi-head attention mechanism is effective at extracting global features, and the CNN in the embedding block is effective at extracting local features. The accuracy is very close for both approaches, yet CNN shows better precision, recall, and F-score. The improvements, though small, indicate a more consistent balance among false positives and false negatives when CNN is utilized for patch embedding.

The proposed model is evaluated for various combinations of the model parameters. Table VII and Figure 13 present the performance comparison of the proposed ArrhythTransform (AT) model trained with different hyperparameter settings. These models are represented as AT-I to AT-VI. All six models have demonstrated comparable performance with accuracy values exceeding 99% and macro-averaged performances exceeding 97%. Among them, AT V and AT VI have shown the best performance, with AT V (two layers, four heads, patch size of 32, and embedding dimension of 64) achieving the best results with 44,935 trainable parameters. In contrast, lighter variations such as AT I and AT II with 18,471 parameters yield comparable performance but with a slight decline in recall. AT III and IV, with moderate parameter sizes (35,559), provide slight gains but still lag in recall and F-score. Table VIII presents the performance at the beat level. The findings show that the proposed methods have achieved consistent performance in detecting the majority of arrhythmia classes, with precision, recall, and F-scores exceeding 99% for normal beats (N), paced beats (P), and right bundle branch block beats (R). The left bundle branch block (L) and premature ventricular contraction (V) beats trail slightly behind with only minor losses. However, the fusion beat (F) continues to be the class with the lowest performance, with accuracies exceeding 93%, but recall values between 86% and 91%, indicating the challenge in classifying the minority class. The six experimental configurations (AT-I to AT-VI) provide comparable performance with incremental improvements in versions (AT-V and AT-VI), with specific gains in the recognition of F and A classes.

Since the number of heads in the multi-head attention layer and the size of the input patch have a significant influence on the model's performance, the model with an embedding dimension of 64 is further examined with input patch sizes of 8, 16, and 32 and four and eight heads in the attention layer. Figures 14, 15, and Table IX present the statistical results. A patch size of 32 and four heads in the attention layer resulted in the highest model performance. The improvement is apparent with increasing patch size. Accuracy improves from 98.69% (patch size 8, 4 heads) to 99.31% (patch size 32, 4 heads), and the F-score increases from 97.06% to 98.28%. However, the number of attention heads has a limited effect on the performance; for instance, a patch size of 32 yields similar F-scores (98.28% vs. 98.27%) for both 4 and 8 heads. Accuracy is slightly better with fewer attention heads, and recall is mostly consistent across setups. Patch size is therefore a significant contributor; however, adding more heads does not contribute much to classification performance.

Multi-head attention scores are useful for visualizing the detection results. Figure 16 presents the attention scores returned by the transformer encoder's last multi-head attention layer plotted over the input heartbeat signal of classes A, F and L. These attention maps illustrate how several heads were trained to focus on various aspects of the signal in contributing to the final prediction as class F. It can be seen from the attention maps that the transformer model gives higher priority to the middle QRS complex in various heads, demonstrating its clinical importance in heartbeat evaluation. The complementary distribution of attention enables the model to recognize subtle variations in waveform morphology, improving classification performance. This emphasis on physiologically meaningful areas showcases both the accuracy and the interpretability of the model.

Table X presents the performance metrics of numerous studies in the field conducted by researchers. Our proposed model demonstrates superior performance, achieving an accuracy of 99.31% in categorizing seven categories of cardiac arrhythmia beats.

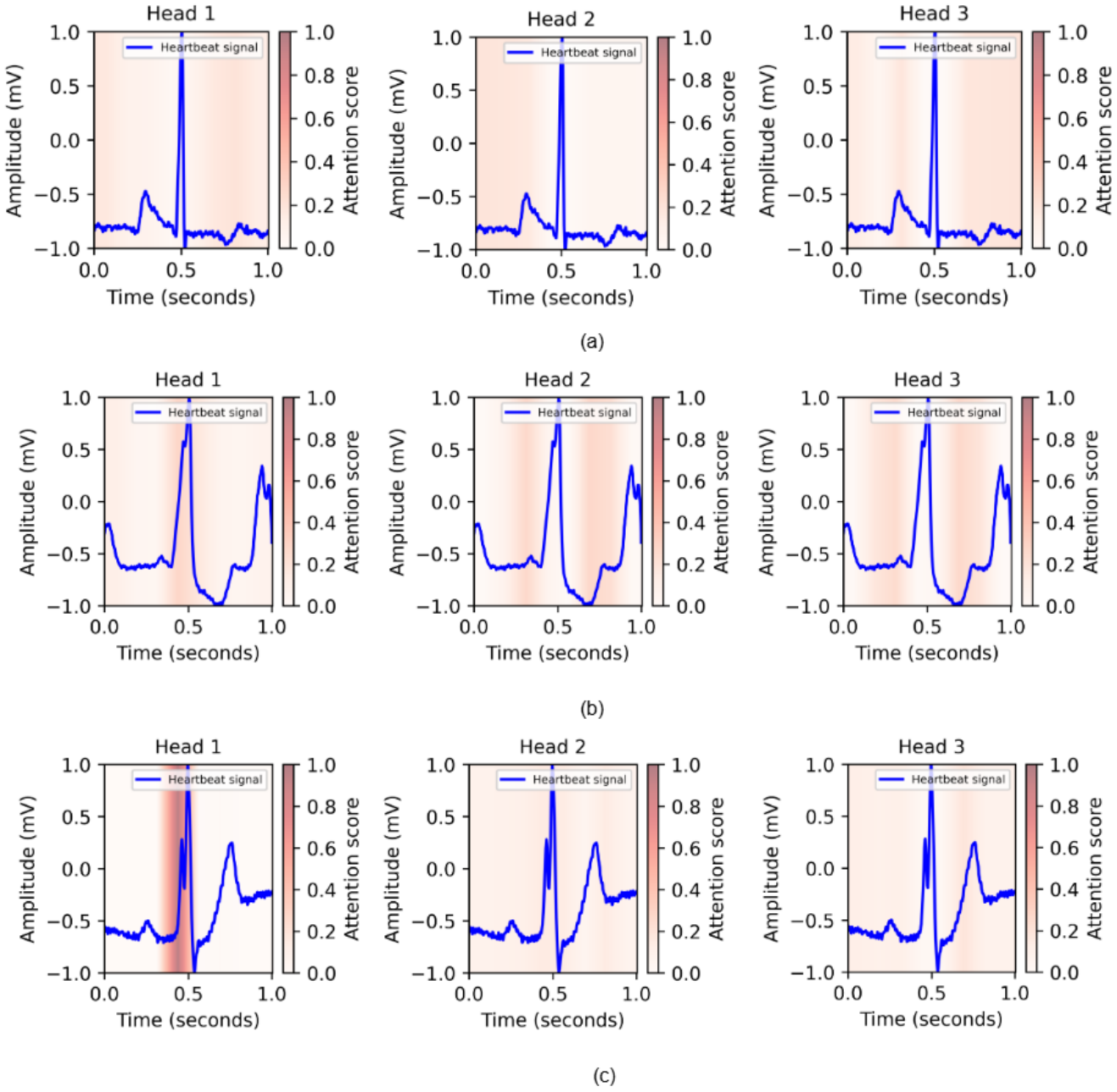

Fig. 16. Visualization of attention maps from the first three heads for the classes (a) A, (b) F, and (c) L.

TABLE X
COMPARISON WITH STATE-OF-THE-ART MODELS

Study	Method	Precision (%)	Recall (%)	F-score (%)	Accuracy (%)
Ozal Yildirim et al. [3]	1D CNN	92.52	93.52	92.45	95.2
Amin Ullah et al. [23]	2D CNN	98.69	97.26 (sen)	0.98	98.92
Muhammad Salman Haleem et al. [20]	CNN + BiLSTM	–	–	–	97.9%
Wei Zeng et al. [19]	CNN + LSTM	–	–	–	97.20
Yong Xia et al. [33]	AE + Transformer	–	–	–	97.93
Taymaz Akan et al. [30]	Transformer Encoder	95.00	80.00	86.00	98.00
YanYun Tao et al. [29]	Refined attention Transformer	96.5	97.6	97.1	–
Proposed Model	ArrhythTransform	**98.41**	**98.16**	**98.28**	**99.31**

IV. CONCLUSION

In this work, a transformer encoder-based architecture with a multi-head attention module ArrhythTransform is proposed for the automatic classification of arrhythmia beats from ECG signals. The performance of the model is evaluated using the MIT-BIH arrhythmia database. A comprehensive performance assessment of the proposed model was con-

ducted for key combinations of hyperparameters, including the number of heads (4, 8), layers (2, 4), and the patch embedding dimension (32, 64) with different patch sizes (8, 16, and 32). Furthermore, the impact of CNN and MLP layers in the embedding block on the overall performance of the model was systematically analyzed. The model employing CNN for patch embedding demonstrated the highest average accuracy with an input patch size of 32, four heads in the MHA layer, and an embedding dimension of 64. The architecture effectively classified seven types of arrhythmia beats with an F-score of 98.28%, a precision of 98.41%, a recall of 98.16%, and an overall classification accuracy of 99.31%. The effective global feature extraction of the transformer and the effective local spatial feature extraction of CNN resulted in peak classification performance with reduced complexity. In addition, the attention maps presented help interpret the decision-making process of the model. As illustrated by the comparison analysis, the proposed ArrhthTransform model achieves performance superior to the state-of-the-art techniques. The proposed ArrhthTransform model, with its lower complexity and interpretable attention maps, is well-suited for real-time arrhythmia detection and can be extended to clinical deployment.

REFERENCES

[1] M. Vaduganathan, G. Mensah, J. Turco *et al.*, "The global burden of cardiovascular diseases and risk: A compass for future health," *Journal of the American College of Cardiology (JACC)*, vol. 80, no. 25, pp. 2361–2371, Dec. 2022.

[2] X. Dong and W. Si, "Heartbeat dynamics: A novel efficient interpretable feature for arrhythmias classification," *IEEE Access*, vol. 11, pp. 87 071–87 086, 2023.

[3] Ö. Yıldırım, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ecg signals," *Computers in biology and medicine*, vol. 102, pp. 411–420, 2018.

[4] S. Boda, M. Mahadevappa, and P. K. Dutta, "An automated patient-specific ecg beat classification using lstm-based recurrent neural networks," *Biomedical Signal Processing and Control*, vol. 84, p. 104756, 2023.

[5] M. Baygin, T. Tuncer, S. Dogan, R. Tan, and U. R. Acharya, "Automated arrhythmia detection with homeomorphically irreducible tree technique using more than 10,000 individual subject ecg records," *Information Sciences*, vol. 575, pp. 323–337, 2021.

[6] J. Ritter, X. Chen, L. Bai, and J. Huang, "Predicting hypotension by learning from multivariate mixed responses," in *Proceedings of The International MultiConference of Engineers and Computer Scientists 2023*, 2023, pp. 1–6.

[7] W. Wang, N. Kumar, J. Chen, Z. Gong, X. Kong, W. Wei, and H. Gao, "Realizing the potential of the internet of things for smart tourism with 5g and ai," *IEEE Network*, vol. 34, no. 6, pp. 295–301, 2020.

[8] V. Rajinikanth, S. Yassine, and S. A. Bukhari, "Hand-sketchs based parkinson's disease screening using lightweight deep-learning with two-fold training and fused optimal features," *International Journal of Mathematics Statistics and Computer Science*, vol. 2, pp. 9–18, 2023.

[9] ingshi Zhang, Y. Zhang, and J. Zhou, "Research on lightweight infrared target detection algorithm based on deep learning," *Engineering Letters*, vol. 33, no. 7, pp. 2589–2597, 2025.

[10] D. Hareesha, B. Senthilkumaran, K. L. Prasanna, J. V. N. Ramesh, R. Jeyalakshmi, and S. M. Naidu, "Sustainable deep learning-based fault detection for next-generation 6g networks," *IAENG International Journal of Computer Science*, vol. 52, no. 7, pp. 2508–2517, 2010.

[11] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, and R. S. Tan, "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, vol. 89, pp. 389–396, 2017.

[12] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ecg classification by 1-d convolutional neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 3, pp. 664–675, 2015.

[13] N. Sabor, G. Gendy, H. Mohammed, G. Wang, and Y. Lian, "Robust arrhythmia classification based on qrs detection and a compact 1d-cnn for wearable ecg devices," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 12, pp. 5918–5929, 2022.

[14] A. M. Alqudah and A. Alqudah, "Deep learning for single-lead ecg beat arrhythmia-type detection using novel iris spectrogram representation," *Soft Computing*, vol. 26, no. 3, pp. 1123–1139, 2021.

[15] Y. Zhang, J. Yi, A. Chen, and L. Cheng, "Cardiac arrhythmia classification by time–frequency features inputted to the designed convolutional neural networks," *Biomedical Signal Processing and Control*, vol. 79, p. 104224, 2022.

[16] A. M. Alqudah, S. Qazan, L. Al-Ebbini, H. Alquran, and I. A. Qasmieh, "Ecg heartbeat arrhythmias classification: a comparison study between different types of spectrum representation and convolutional neural networks architectures," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 10, pp. 4877–4907, 2021.

[17] N. Mohebbanaaz, L. V. R. Kumar, and Y. P. Sai, "A new transfer learning approach to detect cardiac arrhythmia from ecg signals," *Signal Image and Video Processing*, vol. 16, no. 7, pp. 1945–1953, 2022.

[18] N. Mohebbanaaz, Y. P. Sai, and L. R. Kumari, "Cognitive assistant deepnet model for detection of cardiac arrhythmia," *Biomedical Signal Processing and Control*, vol. 71, p. 103221, 2021.

[19] W. Zeng, B. Su, Y. Chen, and C. Yuan, "Arrhythmia detection using tqwt, ceemd and deep cnn-lstm neural networks with ecg signals," *Multimedia Tools and Applications*, vol. 82, no. 19, pp. 29 913–29 941, 2022.

[20] M. S. Haleem, R. Castaldo, S. M. Pagliara, M. Petretta, M. Salvatore, M. Franzese, and L. Pecchia, "Time adaptive ecg driven cardiovascular disease detector," *Biomedical Signal Processing and Control*, vol. 70, p. 102968, 2021.

[21] F. S. Butt, M. F. Wagner, J. Schafer, and D. G. Ullate, "Toward automated feature extraction for deep learning classification of electrocardiogram signals," *IEEE Access*, vol. 10, pp. 118 601–118 616, 2022.

[22] E. A. Budisantoso, G. Darmawan, and A. A. Pravitasari, "Improving accuracy with hyperparameter tuning for sarcasm detection in twitter comments using bilstm," *IAENG International Journal of Applied Mathematics*, vol. 55, no. 7, pp. 2042–2050, 2025.

[23] A. Ullah, S. M. Anwar, M. Bilal, and R. M. Mehmood, "Classification of arrhythmia by using deep learning with 2-d ecg spectral image representation," *Remote Sensing*, vol. 12, no. 10, p. 1685, 2020.

[24] A. Isin and S. Ozdalili, "Cardiac arrhythmia detection using deep learning," in *Procedia Computer Science*, vol. 120, 2017, pp. 268–275.

[25] A. Pal, R. Srivastva, and Y. N. Singh, "Cardionet: an efficient ecg arrhythmia classification system using transfer learning," *Big Data Research*, vol. 26, p. 100271, 2021.

[26] Y. D. Daydulo, B. L. Thamineni, and A. A. Dawud, "Cardiac arrhythmia detection using deep learning approach and time frequency representation of ecg signals," *BMC Medical Informatics and Decision Making*, vol. 23, no. 1, 2023.

[27] L. Meng, W. Tan, J. Ma, R. Wang, X. Yin, and Y. Zhang, "Enhancing dynamic ecg heartbeat classification with lightweight transformer model," *Artificial Intelligence in Medicine*, vol. 124, p. 102236, 2022.

[28] R. Hu, J. Chen, and L. Zhou, "A transformer-based deep neural network for arrhythmia detection using continuous ecg signals," *Computers in Biology and Medicine*, vol. 144, p. 105325, 2022.

[29] Y. Tao, B. Xu, and Y. Zhang, "Refined self-attention transformer model for ecg-based arrhythmia detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–14, 2024.

[30] T. Akan, S. Alp, and M. A. N. Bhuiyan, "ECGFoRMer: Leveraging Transformer for ECG Heartbeat Arrhythmia Classification," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2023.

[31] H. El-Ghaish and E. Eldele, "ECGTransForm: Empowering adaptive ECG arrhythmia classification framework with bidirectional transformer," *Biomedical Signal Processing and Control*, vol. 89, p. 105714, 2023.

[32] D. Wang, L. Qiu, W. Zhu, Y. Dong, H. Zhang, Y. Chen, and L. Wang, "Inter-patient ECG characteristic wave detection based on convolutional neural network combined with transformer," *Biomedical Signal Processing and Control*, vol. 81, p. 104436, 2022.

[33] Y. Xia, Y. Xiong, and K. Wang, "A transformer model blended with CNN and denoising autoencoder for inter-patient ECG arrhythmia classification," *Biomedical Signal Processing and Control*, vol. 86, p. 105271, 2023.

[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *arXiv preprint arXiv:1706.03762*, 2017. [Online]. Available: https://arxiv.org/pdf/1706.03762v5

[35] Y. Wang, G. Yang, S. Li, Y. Li, L. He, and D. Liu, "Arrhythmia classification algorithm based on multi-head self-attention mechanism," *Biomedical Signal Processing and Control*, vol. 79, p. 104206, 2022.

[36] G. Moody and R. Mark, "The impact of the MIT-BIH Arrhythmia Database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[37] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.