Leveraging Visibility Graphs for Enhanced Arrhythmia Classification with Graph Convolutional Networks

Rafael F. Oliveira[1], Gladston J. P. Moreira[1], Vander L. S. Freitas[1], Eduardo J. S. Luz[1]

[1]Computer Department, Federal University of Ouro Preto, 122, Diogo de Vasconcelos Street, Pilar, Ouro Preto, 35402163, Minas Gerais, Brazil.

Contributing authors: rafael.fo@aluno.ufop.edu.br; gladston@ufop.edu.br; vander.freitas@ufop.edu.br; eduluz@ufop.edu.br;

Abstract

Arrhythmias, detectable via electrocardiograms (ECGs), pose significant health risks, emphasizing the need for robust automated identification techniques. Although traditional deep learning methods have shown potential, recent advances in graph-based strategies are aimed at enhancing arrhythmia detection performance. However, effectively representing ECG signals as graphs remains a challenge. This study explores graph representations of ECG signals using Visibility Graph (VG) and Vector Visibility Graph (VVG), coupled with Graph Convolutional Networks (GCNs) for arrhythmia classification. Through experiments on the MIT-BIH dataset, we investigated various GCN architectures and preprocessing parameters. The results reveal that GCNs, when integrated with VG and VVG for signal graph mapping, can classify arrhythmias without the need for preprocessing or noise removal from ECG signals. While both VG and VVG methods show promise, VG is notably more efficient. The proposed approach was competitive compared to baseline methods, although classifying the S class remains challenging, especially under the inter-patient paradigm. Computational complexity, particularly with the VVG method, required data balancing and sophisticated implementation strategies. The source code is publicly available for further research and development at https://github.com/raffoliveira/VG_for_arrhythmia_classification_with_GCN.

Keywords: Arrhythmia classification, Visibility graphs, GCN, ECG.

1 Introduction

Early detection of heart diseases is essential to enable preventive measures. The electrocardiogram (ECG) [1] is the primary diagnostic tool for heart conditions. Arrhythmias are heart issues detectable via ECG that impair the heart's ability to pump sufficient blood, potentially affecting the brain and other organs. Arrhythmias can

1

be life-threatening, necessitating continuous cardiac activity monitoring for appropriate medical response [2, 3].

The intricate and challenging task of manually identifying and classifying arrhythmias highlights the essential need for further research into automated solutions. Techniques involving Machine Learning, such as Artificial Neural Networks (ANNs) and Deep Neural Networks (DNNs), have been increasingly adopted [4–6]. However, achieving stringent and rigorous outcomes remains challenging, particularly within the inter-patient paradigm, where training and testing sets include distinct patient data [4, 7, 8]. In contrast, the intra-patient paradigm may encounter overestimated outcomes when training and testing sets consist of data from the same patient. De Chazal et al. [8] have articulated the importance of standard protocols for the sake of comparability among published studies, highlighting that those not adhering to such benchmarks might not fully realize their practical potential and could present results with a degree of bias [4, 9].

Numerous neural network-based methodologies proposed in the literature aim to address the automated classification of arrhythmias. [10, 11]. Studies such as those conducted by Essa and Xie [11] and Mousavi and Afghah [10] have shown significant success in adhering to the AAMI [7] standards and the inter-patient paradigm. Both used Convolutional Neural Networks (CNNs) in their machine learning models, albeit with distinct approaches. Essa and Xie [11] proposed a hybrid architecture that combined CNN-LSTM and LSTM networks using heartbeat segments, whereas Mousavi and Afghah [10] introduced a Recurrent Neural Network (RNN) structure consisting of an encoder-decoder with a CNN, applied to sequences of heartbeats. Garcia et al. [12] even introduced an innovative approach utilizing a temporal vectorcardiogram with feature selection through complex networks paired with an SVM classifier. When tested under the inter-patient paradigm, this method achieved an overall accuracy of 92.4%

Despite these advancements, exploring and representing ECG signals as graphs and applying Graph Neural Networks (GNNs) to this representation remains an underexplored area. This gap presents an opportunity for our proposal: leverage the Visibility Graph algorithm to transform ECG data into a graph structure, enabling the use of GNNs. This approach combines the inherent advantages of graph representation with the robust feature extraction capabilities of GNNs, potentially setting a new benchmark in ECG-based arrhythmia classification.

Focusing on the proposed approach of employing a visibility graph (VG) for ECG signal representation, one question that encapsulates the core objective of the study can be formulated:

- How does the integration of the visibility graph (VG) approach to map ECG signals into graph structures and the incorporation of multi-lead ECG data in vector visibility graph (VVG) representations influence the classification performance and diagnostic accuracy of graph neural networks in identifying cardiac arrhythmias within the inter-patient evaluation protocol?

Interestingly, our results show that simpler GCN architectures are more effective than their complex counterparts, emphasizing efficiency and the accurate capture of ECG signal morphology. The study also highlights the significant impact of the inter-patient and intra-patient paradigms on classification performance.

2 Background

Network science [13] has proven its value in extracting significant insights from various domains, including time series and sequential data, enabling the characterization of nonlinear dynamic behavior in diverse contexts. Researchers have proposed several approaches for determining the spatial connectivity of data over time through complex networks. As highlighted by Ren and Jin [14], there are three primary methodologies prominent in the literature:

- Mapping of sequential data or pseudoperiodic time series into complex networks, where network vertices represent each cycle of the time series. The connectivity between vertices relies on the temporal similarity or correlation between cycles [15, 16];
- Recurrence networks that treat phase space vectors as vertices, with connectivity between vertices determined by the distance of the corresponding vectors [17, 18];

- Direct definition of time series data as network vertices, where connectivity among vertices is determined based on the temporal sequence of data points such as a visibility criteria [19–21] or temporal neighborhood [22].

Among the commonly used approaches in the literature, this article focuses on mapping ECG signals into graphs using two methods: the VG method proposed by Lacasa et al. [19] for univariate time series (ECG signals with one lead) and the VVG method proposed by Ren and Jin [14] for multivariate time series (ECG signals with two leads).

2.1 Visibility Graph (VG)

The samples from an ECG lead curve over time can be conceptualized as a one-dimensional time series, which can then be transformed into a graph using the Visibility Graph (VG), as depicted in Figure 1. In this representation, each point in the series becomes a node in the graph, and two nodes are connected if they satisfy a visibility criterion, as follows: given two points, (t_a, y_a) and (t_b, y_b), represented by vertices a and b, where t_a and t_b represent time and y_a and y_b their associated values, a connection is established between them if

$$y_c < y_b + \left(y_a - y_b\right)\frac{t_b - t_c}{t_b - t_a} \qquad (1)$$

for any points (t_c, y_c). In simpler terms, if each point in the series is a vertical bar in a bar chart, two bars a and b would connect if there is no bar c between them whose height would prevent drawing a straight line between the peaks of a and b.

According to Lacasa et al. [19], the graph generated using the Visibility Graph (VG) method consistently demonstrates certain intrinsic features. First, it is a connected graph wherein each vertex maintains visibility to its immediate neighbors, encompassing both left and right adjacencies. Second, the graph is undirected, a characteristic inherent to the algorithm's construction. Finally, the graph maintains invariance under data transformations of the series, with the visibility criterion remaining unaffected by horizontal and vertical axis rescaling and translations.

2.2 Vector Visibility Graph (VVG)

Expanding on the Visibility Graph (VG) method for univariate time series, Ren and Jin [14] proposed the Vector Visibility Graph (VVG) method for mapping multivariate time series into a directed complex network. In this approach, a multidimensional data vector is defined as a node, and the connectivity between nodes is established based on the visibility criterion applied to the corresponding data vectors.

Consider $X_t = \{x_t^i\}_{i=1}^m$, a multidimensional m-dimensional[1] time series, where N is the size of each dimension and $\vec{X}_t$ is a vector representing the space of the multivariate time series, given by $\vec{X}_t = [x_t^1, x_t^2, ..., x_t^m]$. For any two vectors $\vec{X}_a$ and $\vec{X}_b$ in the vector space $\vec{X}_t$, the projection of $\vec{X}_a$ onto $\vec{X}_b$ can be defined as follows:

$$||\vec{X}_b^a|| = \frac{\sum_{i=1}^m x_a^i x_b^i}{\sqrt{\sum_{i=1}^m x_a^i x_a^i}}. \qquad (2)$$

Assuming that each vector in the vector sequence is represented as a node in the network, the visibility criterion between vectors is defined as follows:

$$||\vec{X}_c^a|| < ||\vec{X}_b^a|| + \left(||\vec{X}_a|| - ||\vec{X}_b^a||\right)\frac{t_b - t_c}{t_b - t_a}, \qquad (3)$$

where $t_a < t_c < t_b$, $||\vec{X}_b^a||$ represents the projection of $\vec{X}_a$ onto $\vec{X}_b$, and $||\vec{X}_c^a||$ is the projection of $\vec{X}_a$ onto $\vec{X}_c$. Thus, if the criterion (3) is fulfilled, a connection will be established from the vertex represented by $\vec{X}_a$ to the vertex represented by $\vec{X}_b$ in the resulting complex network. When $m = 1$, the VVG method becomes equivalent to the VG method. Figure 2 demonstrates the application of the VVG method. In this illustrated example, multivariate time series data, such as dual-lead ECG signals, are represented by a graph structure in which each series point corresponds to a network node. The visibility between these points, as determined by the VVG criteria, is illustrated to show how connections are formed within the graph. This representation highlights the complex relationships within the multivariate data and underscores the

---

[1] Multidimensional time series representation for m dimensions: $X_t = [\{x_t^1, x_{t+1}^1, x_{t+2}^1, ..., x_{t+n}^1\}, \{x_t^2, x_{t+1}^2, x_{t+2}^2, ..., x_{t+n}^2\}, ..., \{x_t^m, x_{t+1}^m, x_{t+2}^m, ..., x_{t+n}^m\}]$
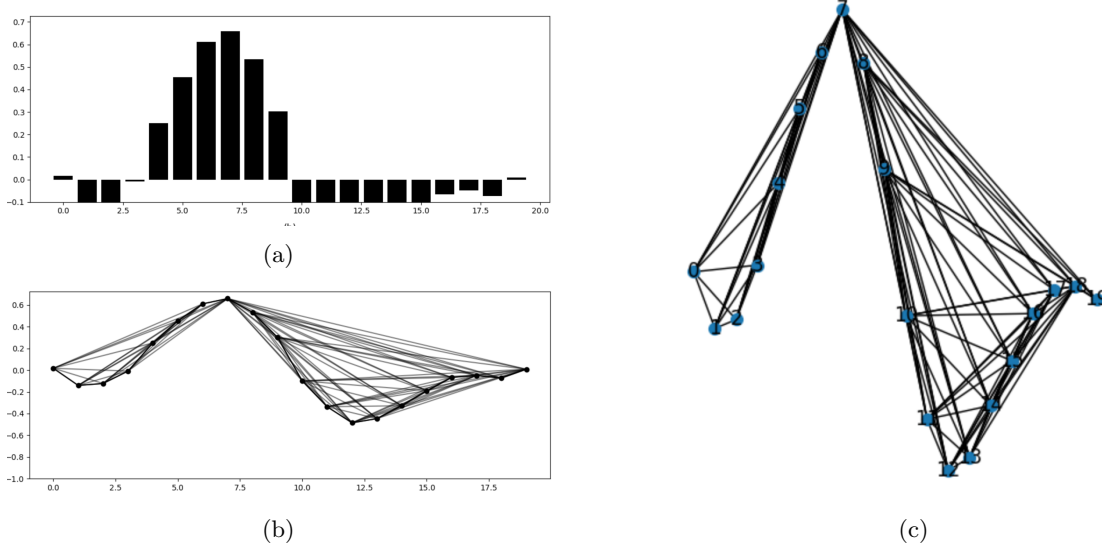
Fig. 1: Example of the VG method application. (a) Visibility among the points of a univariate time series (ECG signal with one lead). (b) Graph generated by VG. (c) Graph visualization.

potential of the VVG method in capturing complex patterns in biomedical signals like ECG.

2.3 Rationale for using VG and VVG

The rationale behind employing the Visibility Graph (VG) methodology for analyzing electrocardiogram (ECG) signals in our application is motivated by the nature of the ECG waveforms. ECG signals are characterized by prominent and well-defined waves, notably the P wave, QRS complex, and T wave, reflecting specific electrical activities within the heart. These waves form patterns that can indicate various cardiac conditions, including arrhythmias.

The hypothesis driving this approach is that in a normal cardiac rhythm, especially observable in lead II of the ECG, the peaks of these waves (P, QRS, and T) will exhibit a specific pattern of interconnectivity. The VG method transforms these time-series data points into a network of vertices (representing the wave peaks) and edges (representing the visibility or direct line of sight between these peaks). One expects to observe a consistent pattern of connections between these peaks in a regular heartbeat.

Conversely, these patterns will likely deviate from the norm in an arrhythmic beat. Arrhythmias often manifest irregularities in the timing,

sequence, and morphology of ECG waves. For instance, atrial fibrillation may be characterized by an irregular rhythm and the absence of P waves, whereas ventricular tachycardia may show aberrant QRS complexes. By applying the VG method, we hypothesize that these disruptions in the waveform will result in distinctly different graph structures compared with those derived from normal heartbeats. This difference in graph topology is investigated through graph neural networks (GNNs).

The VG approach offers a way to encapsulate the intrinsic characteristics of ECG waveforms into a graph structure, providing another perspective for ECG analysis. By translating the ECG signal into a visibility graph, we capture the individual features of each wave and the contextual relationships between them. This could enhance our ability to detect and classify cardiac arrhythmias more accurately, leveraging the complex interplay of ECG wave characteristics that are otherwise challenging to discern through traditional time-series analysis methods.

2.4 Graph Neural Networks

The emergence of Graph Neural Networks (GNNs) dates back to the 1990s [23], starting with recursive neural networks [24] applied to directed
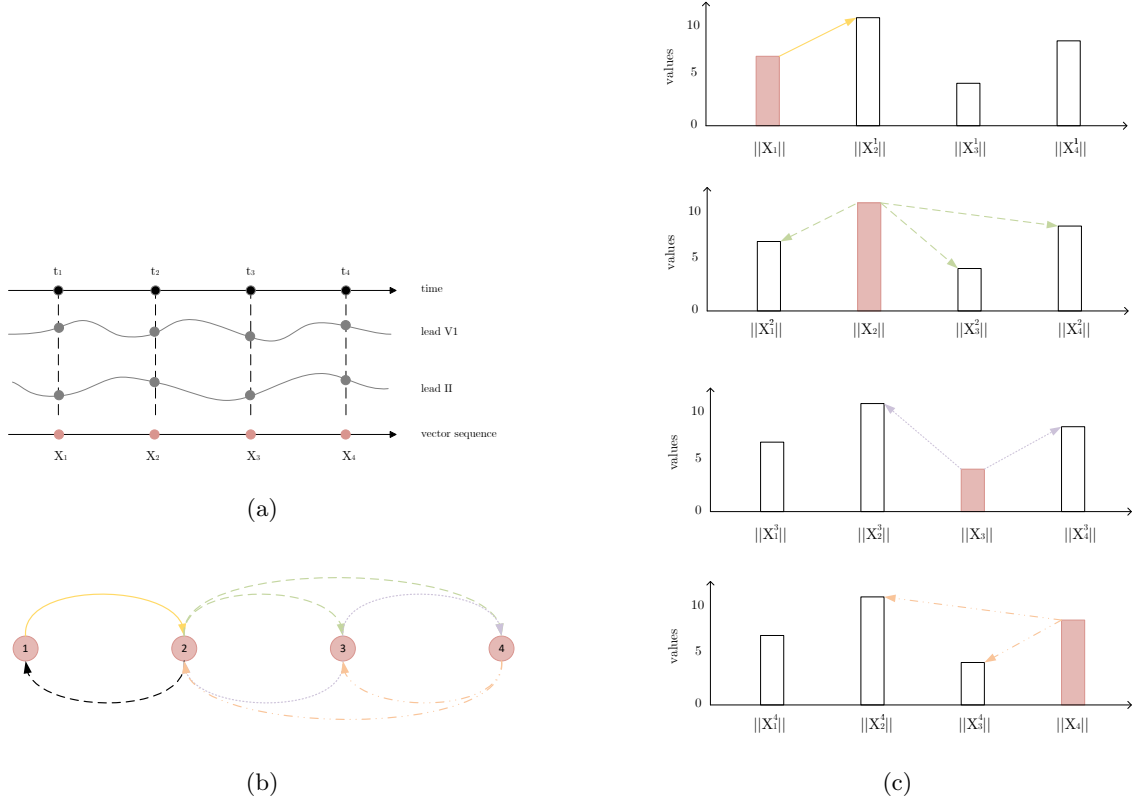
4

Fig. 2: Example of the VVG method application. (a) Multivariate time series represented as two ECG signals from different leads. (b) Graph generated by the VVG. (c) Visibility among the vectors. Source: Adapted from Ren and Jin [14].

acyclic graphs. Subsequently, Recurrent Neural Networks (RNNs) [25] and Feed-Forward Neural Networks (FFNNs) [26] are introduced for cyclic graphs. With the advancement of deep learning, mainly through Convolutional Neural Networks (CNNs) [27], GNNs have also undergone significant enhancements.

According to Wu et al. [28], GNNs can be categorized into four main types: Recursive GNNs (RecGNN), Graph Autoencoders (GAE), Spatial-Temporal GNNs (STGNN), and Convolutional GNNs (ConvGNN) or Graph Convolutional Networks (GCN). Convolutional GNNs (ConvGNNs) or Graph Convolutional Networks (GCNs) are the primary approaches for automatic arrhythmia classification through mapping ECG signals onto graphs in this work.

Convolutional GNNs (ConvGNNs) extend the concept of convolutional operations from the Euclidean domain, typically represented in grid formats, to graph-structured data. The core idea is to generate a representation for a node v by aggregating its attributes $\mathbf{x}_v$ and the attributes of its neighboring vertices $\mathbf{x}_u$, where $u \in N(v)$, in which $N(v)$ is the set of neighbors.

ConvGNNs employ multiple convolutional layers to extract high-level representations of nodes. This means that a single convolutional layer aggregates information from the first-order neighbors of a node, two layers aggregate information from two-hop neighborhoods, and so on. Consequently, the more convolutional layers used, the more extensive the information aggregated from a node's neighbors. Thus, ConvGNNs play a central role in developing complex GNN models because they effectively capture and integrate local and extended neighborhood information [28].

As detailed by Wu et al. [28], Convolutional Graph Neural Networks (ConvGNNs) are primarily distinguished into two methodologies. The first, spectral-based, defines graph convolutions by introducing filters in the realm of graph signal processing [29], conceptualizing graph convolution as a means to remove noise from graph signals. The spatial-based one builds upon the propagation concept from Recursive GNNs (RecGNNs) and focuses on defining graph convolutions through disseminating information across neighborhoods. This latter approach has seen rapid development due to its efficiency, flexibility, and general applicability [30].

In this study, we chose the spatial-based methodology for Convolutional Graph Neural Networks (ConvGNNs) as the foundation for our proposed automatic ECG signal classification method. This decision is grounded in several key factors that align with the specific requirements and characteristics of ECG data analysis, such as (i) localized feature learning because ECG signals exhibit localized features, such as particular waveforms and intervals, that are crucial for identifying arrhythmias. In addition, (ii) convolution operations are highly efficient. ECG-derived graphs can exhibit diverse and complex connectivity patterns, reflecting the intricate nature of cardiac electrical activity. The spatial approach allows for efficient processing of these graphs, adapting to the unique topology of each ECG-derived graph without the need for complex spectral transformations. (iii) The spatially-based ConvGNNs are robust to graph structure and size variations, making them suitable for ECG signals, which can vary significantly between individuals. This robustness ensures the model can generalize well across different patients, enhancing its utility in real-world clinical settings. (iv) Spatial-based ConvGNNs operate by directly aggregating features from neighboring nodes, potentially offering clinicians more intuitive interpretations.

Graph Convolutional Networks (GCNs) receive as input a graph that is represented as $G(V, E)$, where V is the set of vertices and E the set of edges, with each vertex connected to itself to include its attributes in aggregation with its neighbors' attributes. The attribute matrix $X \in \mathcal{R}^{n \times d}$ contains attribute vectors for each vertex. The adjacency matrix A and the diagonal degree matrix D, with $D_{ii} = \sum_j A_{ij}$, are modified by adding the identity matrix to A for loops, resulting in

$\tilde{A} = A + \lambda I_n$. A GCN with multiple layers captures information from a wider range of neighborhoods, following the propagation rule [30]:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}), \qquad (4)$$

where $\tilde{A}$ is the adjacency matrix with loops, $\tilde{D}$ the degree matrix with loops, $W^{(l)}$ the weight matrix, and σ the activation function, such as ReLU, with $H^{(0)} = X$.

Because the number of neighbors of a vertex can vary from one to thousands, it is sometimes inefficient to consider the entire neighborhood. For this problem, a network called GraphSAGE [31] samples a fixed number of neighbors for each vertex. The convolution operation is given by:

$$h_v^{(l)} = \sigma(W^{(l)} \cdot f_l(h_v^{(l-1)}, \{h_u^{(l-1)}, \forall u \in S_{N(v)}\})), \qquad (5)$$

where f is an aggregation function, σ the activation function, $S_{N(v)}$ the sampled neighbor of vertex v and $h_v^{(l)} \in \mathcal{R}^d$ the attribute vector of vertex v of the l-th layer with $h_v^{(0)} = X_v$. GraphSAGE aggregates information from local neighbors, enhancing the classification process with each iteration. The aggregated information is concatenated and normalized in each iteration, incrementally enriching the vertex representation.

GNNs analyze different levels of graph tasks, as outlined by Zhou et al. [23]. Node-level tasks include classification, regression, and clustering of vertices. Edge-level tasks involve classifying or predicting edges, whereas graph-level tasks encompass classification and regression for the entire graph. In this study, heartbeats are mapped into graphs, and the graph-level learning mechanism classifies each heartbeat by considering the graph as a whole.

3 Related Works

This section summarizes studies from 2017 to 2023 that explored deep learning techniques for the problem of arrhythmia classification in ECG signals, as seen in Table 1.

Hannun et al. [6] introduces a landmark development in arrhythmia classification through ECG signals, utilizing a basic CNN architecture. This study stands out because it uses a large private dataset encompassing ECGs from 53,549 patients, demonstrating that deep learning techniques can

Table 1: State-of-the-art DL works for arrhythmia classification.

Reference	Dataset	# Classes	Method	Performance (%)	AAMI	Inter-patient
Cao et al. [32]	MIT-BIH	4	Pre-trained ResNet18	$Acc^* \Rightarrow 90.8$, $Pr_N \Rightarrow 95.3$, $Re_N \Rightarrow 95.1$ $Pr_S \Rightarrow 13.0$, $Re_S \Rightarrow 9.0$, $Pr_V \Rightarrow 68.2$, $Re_V \Rightarrow 88.4$, $Pr_F \Rightarrow 1.3$, $Re_F \Rightarrow 0.3$	$\pm^*$	✓
Gai [33]	MIT-BIH	5	Pre-trained ImageNet Wigner-Villee distribution	$Pr^* \Rightarrow 98.62$, $Re^* \Rightarrow 98.65$, $F1^* \Rightarrow 98.62$, $Acc \Rightarrow 98.65$	$\pm$	✓
Essa and Xie [11]	MIT-BIH	4	CNN-LSTM RR-HOS-LSTM	$Acc \Rightarrow 95.81$, $Sp^* \Rightarrow 94.56$, $Se \Rightarrow 69.20$, $F1 \Rightarrow 71.06$, $+P^* \Rightarrow 74.97$, $\kappa^* \Rightarrow 0.79$	✓	✓
Hannun et al. [6]	Own dataset	12	DNN	$AUC^* \Rightarrow 97.0$, $Sp \Rightarrow 75.2$ $F1 \Rightarrow 83.7$	X	✓
Mousavi and Afghah [10]	MIT-BIH	4	Sequence-to-sequence CNN	$Acc \Rightarrow 99.53$, $Se \Rightarrow 96.18$ $+P \Rightarrow 97.2$, $Sp \Rightarrow 98.58$	✓	✓
Garcia et al. [12]	MIT-BIH	3	TVCG* + Complex Networks and SVM	$Acc \Rightarrow 92.4$	✓	✓
Mathews et al. [34]	MIT-BIH	2	Restricted Boltzmann Machines (RBM) Deep Belief Network (DBN)	$Acc \Rightarrow 95.2$, $Se \Rightarrow 80.5$ $+P \Rightarrow 47.97$, $FPR \Rightarrow 4.65$	✓	✓

$^*Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $Sp \Rightarrow$ Specificity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $Pr \Rightarrow$ Precision, $Re \Rightarrow$ Recall, $F1 \Rightarrow$ F1-score, $AUC \Rightarrow$ Area Under the Curve, $\kappa \Rightarrow$ Kappa, $\pm \Rightarrow$ Partially met AAMI standard, $TVCG \Rightarrow$ Temporal Vectorcardiogram.

surpass even cardiologists in arrhythmia detection and classification. The innovative approach of using single-channel ECG signals without extensive preprocessing highlights the effectiveness of deep learning despite data and signal limitations. However, the dataset's exclusivity limits the results' replicability, in contrast with the ANSI/AAMI [7] norm recommending the use of public datasets, such as the widely recognized MIT-BIH, for more standardized and comparable evaluations.

Studies employing the MIT-BIH Arrhythmia Database for evaluation, such as in [11, 32–34], are distinguished for adhering to the ANSI/AAMI standard, ensuring reproducibility and fair comparisons with other techniques. Nevertheless, De Chazal et al. [8] identified a lack of standardization in using the MIT-BIH, particularly when not employing an inter-patient scheme for evaluation. Intra-patient schemes, such as cross-validation, may yield clinically unrealistic results. For example, Gai [33] does not strictly follow the De Chazal et al. protocol and uses a limited number of beats for testing, making comparisons with other methods challenging. This lack of standardization and non-adherence to the protocol underscores the need to reevaluate and reimplement other methods for a fair comparison.

State-of-the-art studies following the De Chazal et al. protocol and the ANSI/AAMI standards include [10, 11, 32, 34]. These studies are distinguished by using deep learning in a fair and robust approach, employing an inter-patient scheme in constructing training and test datasets. For instance, Mousavi and Afghah [10] study used a sequence-to-sequence approach, processing multiple heartbeats simultaneously, which is crucial for inferring cardiac rhythm from a chain of beats. The model combines Convolutional Neural Networks (CNN) for feature extraction and Recurrent Neural Networks (RNN) with LSTM units for encoding and decoding these sequences. The results demonstrate the proposed model's superior performance compared with other algorithms, achieving high accuracy, sensitivity, and positive predictive value for the analyzed arrhythmia categories in both intra-patient and inter-patient paradigms.

In [34], a novel approach is proposed for the classification of ECG signals using deep learning, explicitly employing Restricted Boltzmann Machines (RBM) and Deep Belief Networks (DBN) for detecting ventricular and supraventricular arrhythmias. This study utilizes the MIT-BIH database and conducts preprocessing of ECG signals through filtering artifacts such as baseline wander, power line interference, and high-frequency noise. The authors implemented two feature extraction techniques to produce feature sets, including RR intervals, heartbeat intervals, and segmented morphology. The results demonstrate high accuracy in detecting ventricular ectopic beats (93.63%) and supraventricular ectopic beats (95.57%).

In [11], an automatic system for cardiac arrhythmia classification is proposed using deep learning models that combine CNN (Convolutional Neural Network) and LSTM (Long Short-Term Memory) to capture local features and temporal dynamics in ECG data. This model integrates classic features such as RR intervals and Higher Order

Statistics (HOS) with the LSTM model to high-light abnormal heartbeat classes effectively. These models are trained on different data subsets to address class imbalance and then combined using a meta-classifier. Another model further verifies the outcome of the meta-classifier to reduce false positives. Experimental results obtained from the MIT-BIH Arrhythmia Database and following a "subject-oriented (a.k.a. intra-patient)" independent patient evaluation scheme revealed that the proposed method achieves an overall accuracy of 95.81%. The average F1 score and positive predictive value exceeded all other methods by over 3% and 8%, respectively.

The study in [32] introduces a method for cardiac arrhythmia classification using deep learning with the ResNet-18 model. The preprocessing of the MIT-BIH data included filtering to remove noise and segmentation into heartbeats. The method employs a Short-Time Fourier Transform (STFT) to convert 1D ECG signals into 2D time-frequency spectrograms suitable for pre-trained CNN classifiers. Oversampling and undersampling techniques are used to address class imbalance in the dataset. The results showed an accuracy of 90.8%, with high precision and recall for the normal class (N) and varied performance for other types of arrhythmias, such as supraventricular (S), ventricular (V), and fusion (F).

These studies highlight the potential of deep learning when applied in a standardized and rigorous manner, following established protocols to ensure reliable and comparable results in detecting and classifying cardiac arrhythmias. However, as extensively discussed in [4], even methods that evaluate according to the inter-patient protocol can still improve the analysis of results and reproducibility by adding an ablation study in the preprocessing and class imbalance compensation phases. These phases are crucial and can often make more difference than the feature extraction and classification techniques themselves; however, these phases are usually neglected and not well detailed/described in the articles [4].

The present article explores deep learning techniques based on graphs, specifically Graph Neural Networks (GNNs), for which transforming ECG signals and heartbeats into graph representations is essential. Prior research has investigated the transformation of ECG signals into graphs, as done in [12], where a novel ECG representation based on a Temporal Vectorcardiogram (TVCG) is introduced, coupled with a complex network for feature extraction and resource selection using a Particle Swarm Optimization (PSO) algorithm. An SVM classifier is finely tuned in this approach. The proposed method proved effective in the inter-patient paradigm, with results comparable to the state-of-the-art in the MIT-BIH database, achieving 53% positive predictivity for the supraventricular ectopic beat class and 87.3% sensitivity for the ventricular ectopic beat class. The TVCG is a set of points representing two derivations and time in this method. A network is built by considering these points as vertices and the Euclidean distance between each pair of points as edges, forming a square matrix. Initially, each beat is transformed into a regular network (where all vertices are connected), and graph construction occurs through the dynamic removal of edges according to a threshold chosen during training. These graphs extract features, including the Mean connectivity degree, Maximum connectivity degree, Joint degree entropy, Joint degree energy, and Mean joint degree. An SVM classifier is then trained, thus not exploring GNNs for the classification task.

The study in [35] presents an innovative methodology for recognizing abnormalities in 12-lead ECGs using a graph-based neural network (GNN). The methodology comprises two main modules: a feature extractor and a graph neural network module. The feature extractor transforms raw ECG signals into feature vectors, which are used to initialize the graph nodes. The Graph Convolutional Neural Networks (GCN) module then performs convolution and pooling operations on graphs to generate new subgraphs and graph-level representations. For graph construction, each ECG sample is represented by an arbitrary graph with 12 vertices, where each vertex represents a lead. A lead-level attention mechanism was also applied to highlight leads with features most relevant to specific heart diseases. The graph's adjacency matrix describes the connection information between the leads, and the node feature matrix represents the attributes of each lead. The evaluation used the PTB-XL and ICBEB2018 datasets containing 12-lead ECG records with various annotations and diagnostic categories.

In [36], the classification of electrocardiogram (ECG) signals is addressed using an innovative approach that combines edge detection and Graph Neural Networks (GNN). This methodology encompasses two main phases: data transformation and classification. Initially, the ECG is processed as a $64 \times 64 \times 1$ image to construct graph-formatted data. This process includes applying the Sobel operator for edge detection on the curve images. Each pixel with a grayscale intensity value of 128 or higher is converted into a graph node, and the intensity of that pixel becomes an attribute of the node. Edges are employed to connect vertices that group neighboring pixels. A graph is constructed from a single image with nodes and edges derived from the image. The grayscale intensity values of the vertices are normalized for each graph. This normalization involves subtracting the mean of all attributes under each graph from the original value of the preprocessed image, followed by division by the standard deviation. Various graph network architectures are considered for evaluation and conducted on two datasets, the MIT-BIH and PTB-XL. The authors compare their approach with that proposed in [35], claiming superiority as a 100% result is achieved in various graph network architectures. However, inter-patient evaluation is not considered in both approaches proposed in [36] and [35], which may explain the significant results. In addition, recommendations from the ANSI/AAMI EC57 standard for grouping heartbeat classes are not considered.

The evaluation we propose in our study adheres to the ANSI/AAMI EC57 standard and follows the inter-patient evaluation protocol. We also envision conducting experiments without preprocessing the signals or techniques to address class imbalance, as these techniques can significantly influence the final results. By adopting this strategy, we aim to assess the raw and inherent potential of GNNs in discerning arrhythmia patterns directly from ECG data.

Furthermore, we extend this concept by exploring multivariate visibility graphs to incorporate multiple ECG leads. This advancement aligns with clinical practices, where various leads are analyzed for comprehensive cardiac assessment and enriches the data representation, potentially capturing a more complete picture of cardiac rhythm. We have developed an efficient implementation of this multivariate visibility graph method [2] that enhances computational feasibility without compromising the integrity of the ECG signal representation. Compliance with the ANSI/AAMI EC57 standard and the inter-patient evaluation protocol ensures that our findings are not only scientifically rigorous but also hold practical relevance and can be reliably compared with existing methodologies.

4 Experimental methodology

Here, we present the proposed methodology, summarized in Figure 3. No noise-reducing filters are applied to the ECG signals used in their original, unprocessed form.

4.1 Dataset and ANSI/AAMI EC57

Evaluation of automatic arrhythmia classification methods in the literature necessitates a dataset with heartbeats grouped into patient records. The Association for the Advancement of Medical Instrumentation (AAMI) developed a standard, outlined in ANSI/AAMI EC57:1998/(R)2008 [7], to standardize the evaluation of the methods, ensuring reproducibility and comparability. This standard recommends using one of the following five datasets:

- MIT-BIH: The Massachusetts Institute of Technology - Beth Israel Hospital Arrhythmia Database (48 records of 30 minutes each).
- CU: The Creighton University Sustained Ventricular Arrhythmia Database (35 records of 8 minutes each).
- AHA: The American Heart Association Database for Evaluation of Ventricular Arrhythmia Detectors (80 records of 35 minutes each).
- ESC: The European Society of Cardiology ST-T Database (90 records of two hours each).
- NST: The Noise Stress Test Database (12 records of 30 minutes each, plus three records with excessive noise).

The MIT-BIH dataset[3] is most representative in terms of arrhythmia types and is widely used

---

[2]https://github.com/raffoliveira/VG_for_arrhythmia_classification_with_GCN

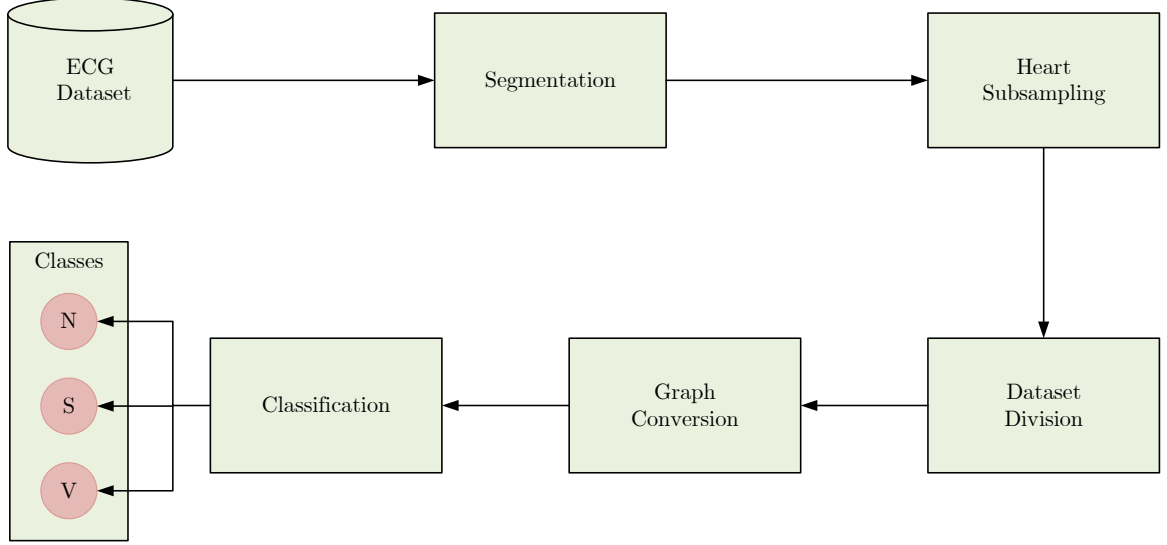[3]https://physionet.org/content/mitdb/1.0.0/

Fig. 3: Flowchart of our proposed methodology.

in literature [37, 38]. This dataset, also used in this work, comprises 48 ECG signal records of 30 minutes from 47 patients sampled at 360 Hz. Each signal contains two leads.

AAMI's standard specifies annotation guidelines for each heartbeat in datasets, recommending excluding pacemaker data, segments with ventricular flutter or fibrillation (VF), and artificial data. We removed data from four patients and grouped them into five main classes: Normal (N), Supraventricular ectopic beat (S), Ventricular ectopic beat (V), Fusion beat (F), and Unknown beat (Q).

AAMI EC57 standard does not specify which data (heartbeats from patients) should be used for training and testing classification models. Data division can follow either an intra-patient or inter-patient paradigm [39]. Intra-patient uses ECG signal data from the same patient for training and testing, whereas inter-patient involves data from different patients without overlap. Using data from the same patient in training and testing often leads to overestimated evaluations [4, 9].

To align tests with real-world scenarios, De Chazal et al. [8] proposed dividing the MIT-BIH dataset into two sets, DS1 and DS2, to ensure no overlap. Table 2 shows the distribution of patient records between the two sets.

Using the inter-patient paradigm and the division proposed by De Chazal et al., Table 3

Table 2: Distribution of MIT-BIH patient records in two sets.

DS1	DS2
101, 106, 108, 109, 112, 114, 115, 116, 118, 119, 122, 124, 201, 203, 205, 207, 208, 209, 215, 220, 223, 230	100, 103, 105, 111, 113, 117, 121, 123, 200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233, 234

summarizes the number of heartbeats per class and their percentage in the MIT-BIH dataset. The table also shows the heartbeat counts after segmentation into training (DS1) and testing (DS2) sets, with percentages of 50,65% and 49,35%. Classes F and Q, with less than 1% presence, are excluded from experiments because of their low occurrence.

4.2 Segmentation

ECG signal segmentation is carried out in sample windows using the R-wave location metadata provided with the MIT-BIH dataset. This involved capturing n points before the R-wave and n points after it, resulting in heartbeats comprising N points, where $N = n_{\text{before}} + n_{\text{after}}$. Each heartbeat is then labeled according to the classes annotated in the dataset. Experiments are conducted to determine this phase's optimal value of n. In some

Table 3: Description of the number of beats in the training and test sets.

Beats	% of total	# Training (DS1)	# Testing (DS2)	# Total
N	89.47%	45844	44238	90082
S	2.76%	944	1837	2781
V	6.96%	3788	3220	7008
F	0.80%	414	388	802
Q	0.01%	8	7	15
Total	**100%**	**50998**	**49690**	**100688**
		50.65%	49.35%	100%

Table 4: Description of sampling strategy.

Beats	Training (DS1)		Testing (DS2)	
	Before	After	Before	After
N	45844	4584	44238	4423
S	944	944	1837	1837
V	3788	3788	3220	3220
Total	**50576**	**7732**	**49295**	**8057**

experiments, ECG signals are also normalized to a range between 0 and 1.

4.3 Heartbeat Subsampling

The preponderance of class N heartbeats, as illustrated in Table 3, necessitates a substantial computational effort due to the numerous graphs required. A subsampling strategy is implemented for class N heartbeats within the training (DS1) and testing (DS2) datasets to facilitate a broader range of experiments. A sampling rate of 10% is adopted, selectively choosing the final heartbeat in every sequence of ten to enhance diversity. Table 4 provides a comparative overview of the dataset pre- and post-balance, including excluding the less prevalent classes F and Q.

4.4 Dataset Division

Heartbeats segmented in the prior stage are divided into a training set (DS1) and a testing set (DS2), as detailed in Table 3. This division adheres to the inter-patient paradigm, aiming to mirror a more realistic scenario [8]. Furthermore, following the subsampling of the predominant class N, as shown in Table 4, it is observed that both the total number of heartbeats and the number in class S are higher in DS2 than in DS1. Consequently, two additional experiments are idealized to examine how GCN models perform when the roles of DS1 and DS2 are reversed and when the data is divided according to the intra-patient paradigm.

4.5 Conversion of Heartbeats into Graphs

In this phase, heartbeats are transformed into graphs using the VG and VVG methods. Figure 4 illustrates the conversion of a heartbeat from the MIT-BIH training set (DS1) corresponding to classes N, S, and V into a graph via the VG method for one ECG lead. This process converts a heartbeat P into a graph $G(V, E)$, where V represents the set of vertices and E the set of edges in a graph G.

This study employs a graph-level approach, where classification considers the graph as a whole. This is similar to Kojima et al. [40], which predicted protein structures using GCN. The use of GNNs enables the incorporation of additional information into the nodes or edges of the graph. Experiments are designed to evaluate different sets of information added to the nodes and their impact on the performance of GCNs in classifying arrhythmias from ECG signals.

4.6 Classification

Five Graph Convolutional Network (GCN) architectures are evaluated for heartbeat classification alongside Convolutional Neural Networks (CNNs) for comparative analysis. While CNNs are not the focus of this study, they provide a benchmark for comparing GCN performance in arrhythmia classification.

In GCN architectures, the input layer takes the attribute vector size (d) of each node ($\mathbf{x}_v \in \mathcal{R}^d$), i.e., the amount of information aggregated at
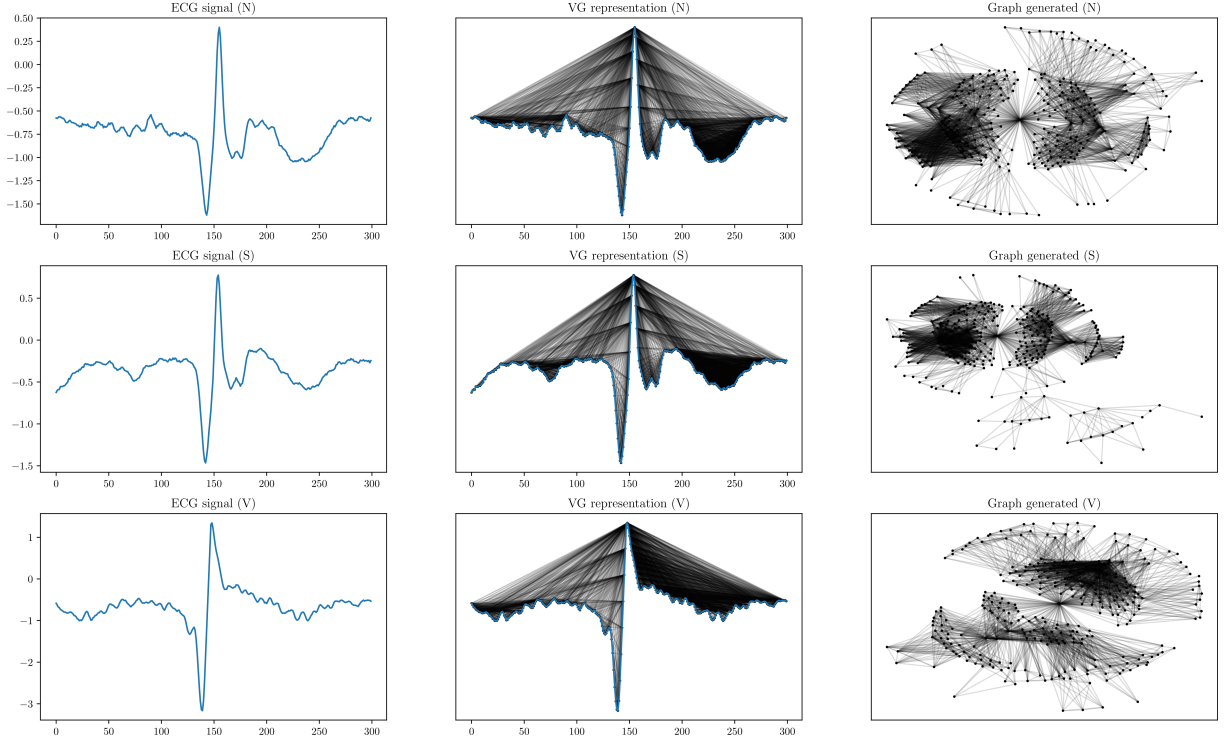
Fig. 4: Example of mapping heartbeats from one lead (patient 118) corresponding to classes N, S, and V using the VG method.

each node. The output layer features three neurons corresponding to the three target classes (N, S, V). For graph-level classification, a *readout* layer aggregates attributes from all vertices in the last iteration of processing, both during training and testing:

$$h_G = \text{readout}(\{h_v^{(l)} | v \in G\}), \qquad (6)$$

where $h_v^{(l)}$ is the attribute vector representation of vertex v at the l-th iteration/layer with $h_v^{(0)} = X$. The *readout* function can be a simple invariant operation like sum or mean or a more sophisticated graph-level aggregation function. Here, the mean operation obtains a high-level representation of the entire graph (h_G) [41].

Following the methodology description, experiments are designed to investigate the primary question of this work: can graph representations of ECG signals using VG and VVG methods enhance arrhythmia classification performance using Graph Convolutional Networks? The designed experiments are as follows:

- **Experiment 1:** Explore new GCN architectures using the VG method alongside various CNN architectures.
- **Experiment 2:** Investigate different window sizes for ECG signal segmentation in the best GCN architectures using the VG method and the optimal CNN architecture.
- **Experiment 3:** Examine information aggregation at graph vertices in the best GCN architectures using VG and VVG methods.
- **Experiment 4:** Investigate the swap of datasets DS1 and DS2 in the best GCN architectures using VG and VVG methods and the optimal CNN architecture.
- **Experiment 5:** Explore the intra-patient paradigm in the best GCN architectures using VG and VVG methods and the optimal CNN architecture.
- **Experiment 6:** Compare the proposed methodology with Garcia et al. [12].

5 Results and Discussion

This section presents the experiments' findings to address the research questions. The experiments encompass the use of VG and VVG methods, as well as the exploration of both inter-patient and intra-patient paradigms and the reversal of data sets. The objective is to analyze whether both graph representations can enhance the performance of arrhythmia classification in ECG signals using Graph Convolutional Networks. The source code is available for reproducibility purposes at https://github.com/raffoliveira/VG_for_arrhythmia_classification_with_GCN.

5.1 Experiment 1: New Architectures

This stage involves experimenting with various GCN and CNN architectures. CNNs serve as a comparative baseline for arrhythmia classification performance across different architectures. The aim is to assess the performance of the proposed GCN architectures against CNNs, which are widely used in the literature. Variations in GCN architectures included the types of layers (traditional graph convolutional layers (GraphConv) and specialized layers (SAGEConv)) and the number of neurons in hidden layers. Variations in CNN architectures focus on the number of convolutional layers.

Table 5 details the proposed GCN architectures for this experiment. Notably, a *readout* layer is used to aggregate vertex attributes in the final iteration of processing to achieve a high-level representation of each entire graph, employing a mean operation. Figures 5, 6, and 7 present the proposed CNN architectures, named CNN-2Conv, CNN-4Conv, and CNN-6Conv, respectively, based on the number of convolutional layers in each architecture. The CNN models employed 1D convolutional layers, with architectures based on an example of CNN use in electroencephalogram signal classification from the Keras library website[4].

For the GCN experiments, each node is enriched with the following information: values from lead V1, values from lead II, and the relative timestamp of each point. This experiment used fewer aggregated data points to analyze different information groups in Experiment 5.3. Both sets of aggregated information are normalized between
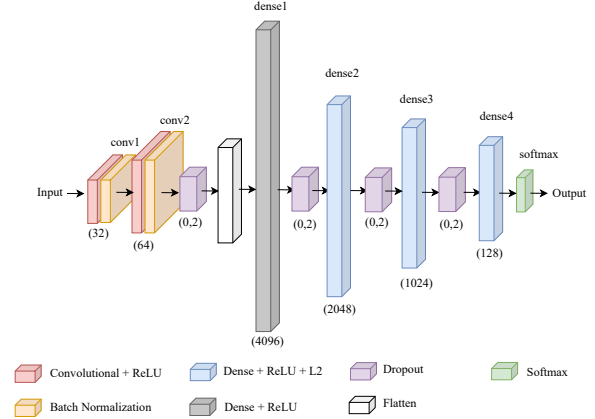
Fig. 5: CNN-2Conv architecture.
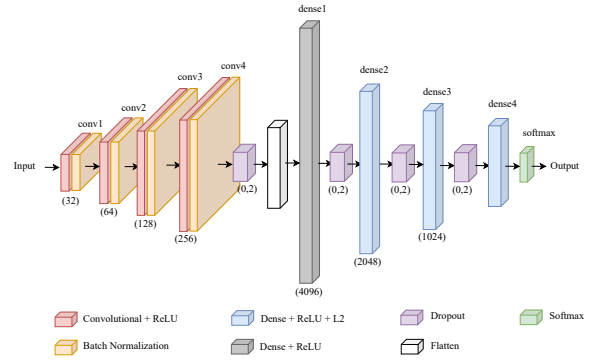
Fig. 6: CNN-4Conv architecture.
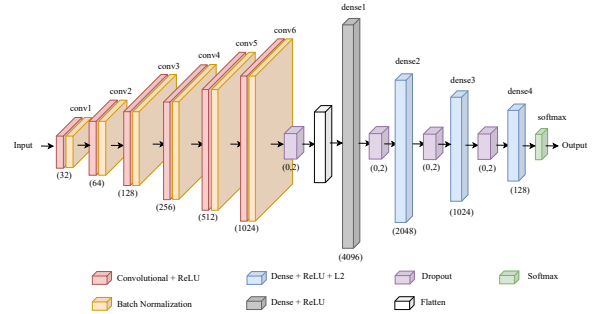
Fig. 7: CNN-6Conv architecture.

$[0, 1]$ using the min-max normalization technique[5]. This technique is chosen to prevent the analyzed architectures from being influenced by extreme value data, as it preserves the order relationships among the data. No additional information is

---

[4]https://keras.io/examples/timeseries/eeg_signal_classification/

[5]$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$

Table 5: GCNs Architectures.

	GCN7			GCN2			GCN60			GCN120			GCN240	
#	Layer	Shape	#	Layer	Shape	#	Layer	Shape	#	Layer	Shape	#	Layer	Shape
1	SAGEConv	$d*\times20$	1	GraphConv	$d\times20$	1	SAGEConv	$d\times60$	1	SAGEConv	$d\times120$	1	SAGEConv	$d\times240$
2	GraphConv	50×40	2	GraphConv	20×3	2	SAGEConv	60×50	2	SAGEConv	120×40	2	SAGEConv	240×140
3	SAGEConv	40×30	4	Readout	-	3	SAGEConv	50×35	3	SAGEConv	40×20	3	SAGEConv	140×40
4	SAGEConv	30×20	5	Softmax	-	4	SAGEConv	35×3	4	SAGEConv	20×3	4	SAGEConv	40×3
5	SAGEConv	20×10				5	Readout	-	5	Readout	-	5	Readout	-
6	GraphConv	10×5				6	Softmax	-	6	Softmax	-	6	Softmax	-
7	GraphConv	5×3												
8	Readout	-												
9	Softmax	-												

*d indicates each vertex's attribute/information vector size.

added to the CNN experiments, leaving the CNNs to "learn" about the ECG signals independently. For both CNN training and converting ECG signals into graphs using the VG method for GNN training, lead II is used. A 10% sampling of class N is performed, where only the last (tenth) heartbeat in a sequence of ten is chosen.

As this experiment focused on parameter adjustments, only the DS1 training set is used, divided into 80% for training (DS1.1) and 20% for validation (DS1.2). The inter-patient paradigm avoids data overlap between training and validation sets, with the distribution of MIT-BIH records as per Table 6. A fixed window of 280 points (100 points before and 180 points after the R peak) is used for segmenting ECG signals based on the state-of-the-art work of [10].

Table 6: Distribution of MIT-BIH records in the DS1 dataset into training (DS1.1) and validation (DS1.2) sets.

Training (DS1.1)	Validation (DS1.2)
101, 106, 108, 112, 115, 116, 118, 119, 122, 124, 201, 203, 205, 208, 209, 215, 220 e 230	109, 114, 207 e 223

In both GCN and CNN training, 150 epochs are used with the Adam optimizer at a learning rate 0.001. For CNN training, the Categorical Cross Entropy function is the loss function. Thus, the CNN models continuously adjust their weights during the loss minimization, leading to better results. The mathematical expression that defines the loss

function is:

$$L(y_{i,\hat{y}_i}) = -\sum_{i=1}^{N} y_i * \log(\hat{y}_i), \qquad (7)$$

where y_i is the actual label during training, $\hat{y}_i$ the predicted label, and N the number of class labels.

Given the results presented in Table 7, the two best GCN architectures and CNN architecture are chosen for further analysis in the following experiments. Table 7 summarizes the performance of architectures on the validation set (DS1.2), with GCN2 and GCN7 architectures showing the best results among GCNs and CNN-2Conv architecture yielding the best result among CNNs. The GCN2 and GCN7 architectures exhibited average F_S of 41% and 54%, respectively. The CNN-2Conv architecture showed an average F_S of 63%. There is significant underperformance in class S for both architectures. As the architectures generally showed low performance, further experiments will analyze other parameters and configurations to improve the current performance.

5.2 Experiment 2: Assessing Segmentation Width for Heartbeat Analysis

This experiment investigated the influence of segmentation width, measured as the number of points per heartbeat, on the efficacy of automatic arrhythmia classification architectures. A critical aspect is that varying the segmentation width could lead to information loss from the ECG heartbeat data.

To this end, three distinct segmentation sizes — 230, 280, and 300 points per heartbeat — are scrutinized, each selected for their prevalence in the literature. The configurations for the dataset

Table 7: Summary of the performance of the GCN and CNN architectures.

Architectures	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	Fs*	+P	Se	FPR	Fs	+P	Se	FPR	Fs	+P	Se	FPR	Fs	
GCN2	47.0	62.0	58.26	53.0	-	-	-	-	40.0	36.0	43.02	38.0	**39.0**	**44.0**	**45.18**	**41.0**	44.0
GCN7	67.0	46.0	18.31	55.0	33.0	1.0	0.26	2.0	55.0	86.0	55.25	67.0	**58.0**	**59.0**	**32.5**	**54.0**	59.0
GCN60	35.0	43.0	64.29	39.0	1.0	1.0	6.52	1.0	32.0	28.0	46.28	30.0	30.0	32.0	50.05	31.0	32.0
GCN120	37.0	47.0	64.64	42.0	1.0	1.0	5.02	1.0	35.0	31.0	44.75	33.0	33.0	35.0	49.37	33.0	35.0
GCN240	36.0	43.0	62.24	39.0	4.0	3.0	8.18	3.0	32.0	29.0	46.99	31.0	31.0	32.0	49.62	31.0	32.0
CNN-2Conv	82.0	50.0	8.89	62.0	43.0	16.0	2.64	23.0	60.0	94.0	48.83	73.0	**70.0**	**65.0**	**25.69**	**63.0**	65.0
CNN-4Conv	77.0	49.0	11.82	60.0	67.0	14.0	0.84	23.0	57.0	90.0	52.6	70.0	67.0	63.0	28.47	60.0	63.0
CNN-6Conv	81.0	43.0	8.16	56.0	79.0	16.0	0.52	26.0	55.0	93.0	58.61	70.0	68.0	62.0	29.41	59.0	62.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

(DS1.1 and DS1.2) and the training protocols mirror those employed in Experiment 5.1.

As tabulated in Table 8, the findings reveal a pronounced advantage at the 280-point segmentation width. In the realm of GCN architectures, GCN7 stood out, registering an average F_S of 54%, while the CNN-2Conv model achieved an average F_S of 63%. Notably, both architectures struggled with classifying class S, highlighting an area ripe for further development. Regarding segmentation width, GCN7 showed an upswing in performance by 20% and 1.87% for average F_S compared to its counterparts at 230 and 300 points, respectively. Similarly, CNN-2Conv demonstrated enhancements of 23.53% and 16.67% in average F_S relative to the 230 and 300-point windows, respectively. Consequently, the 280-point window has been identified as the optimal segmentation width for heartbeat analysis in this study stage.

5.3 Experiment 3: Information Aggregation at Graph Vertices

The capability to aggregate information within graphs, both at vertices and edges, provides an opportunity for incorporating auxiliary and extrinsic data into the graph structure, enhancing the performance of GCNs. This phase assesses the impact of information aggregation on GCN architectures, focusing on extrinsic graph data, as the core purpose of GCNs is to learn from the inherent structures of the graphs used during training. We did not include the CNN model in this experiment.

For clarity in understanding the experiments and the ensuing results, the aggregated vertex information is categorized as follows:

- **II_V1**: Values from lead II, lead V1, and the timing of each point (03 pieces of information).

- **RR**: Values from lead II, lead V1, preceding RR interval, succeeding RR interval, and the timing of each point (05 pieces of information).
- **DifII**: Values from lead II, lead V1, preceding RR interval, succeeding RR interval, the difference between values from lead V1 and lead II, and the timing of each point (06 pieces of information).
- **AvgII**: Values from lead II, lead V1, preceding RR interval, succeeding RR interval, the difference between values from lead V1 and lead II, division of lead V1 values by the mean of lead II, and the timing of each point (07 pieces of information).
- **StdII**: Values from lead II, lead V1, preceding RR interval, succeeding RR interval, the difference between values from lead V1 and lead II, division of lead V1 values by the mean of lead II, division of lead V1 values by the standard deviation of lead II, and the timing of each point (08 pieces of information).
- **Stats**: Values from lead II, lead V1, preceding RR interval, succeeding RR interval, difference between values from lead V1 and lead II, division of lead V1 values by the mean of lead II, division of lead V1 values by the standard deviation of lead II, statistical measures (entropy, variance, standard deviation, mean, median, 5th percentile, 25th percentile, 75th percentile, 95th percentile, RMS, kurtosis, skewness, zero_crossings, mean_crossings) of the values from lead II, and the timing of each point (22 pieces of information).

This categorization provides a comprehensive view of how varying the amount and type of information aggregated at the vertices affects the performance of GCNs in arrhythmia classification.

Table 8: Summary of the GCN and CNN architectures' performance regarding the segmentation width.

Architectures	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	Fs*	+P	Se	FPR	Fs	+P	Se	FPR	Fs	+P	Se	FPR	Fs	
230 points																	
GCN2	43.0	53.0	57.53	47.0	-	-	-	-	37.0	37.0	50.25	37.0	36.0	40.0	48.01	38.0	40.0
GCN7	49.0	45.0	33.63	47.0	5.0	3.0	14.07	4.0	48.0	59.0	48.62	53.0	44.0	46.0	38.04	45.0	46.0
CNN-2Conv	59.0	43.0	24.9	50.0	40.0	15.0	2.7	21.0	50.0	72.0	56.07	59.0	53.0	52.0	36.1	51.0	52.0
280 points																	
GCN2	47.0	62.0	58.26	53.0	-	-	-	-	40.0	36.0	43.02	38.0	**39.0**	**44.0**	**45.08**	**41.0**	44.0
GCN7	67.0	46.0	18.31	55.0	33.0	1.0	0.26	2.0	55.0	86.0	55.25	67.0	**58.0**	**59.0**	**32.3**	**54.0**	59.0
CNN-2Conv	82.0	50.0	8.89	62.0	43.0	16.0	2.64	23.0	60.0	94.0	48.83	73.0	**70.0**	**65.0**	**25.69**	**63.0**	65.0
300 points																	
GCN2	47.0	59.0	56.17	52.0	-	-	-	-	40.0	39.0	44.95	40.0	38.0	43.0	45.18	40.0	44.0
GCN7	72.0	40.0	12.76	52.0	-	-	0.19	-	54.0	92.0	60.75	68.0	56.0	58.0	32.4	53.0	59.0
CNN-2Conv	61.0	54.0	28.97	57.0	40.0	12.0	2.32	19.0	54.0	69.0	46.48	60.0	55.0	56.0	33.7	54.0	56.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

This experiment's dataset configurations and training settings align with those used in Experiment 5.1. The key difference in this phase is the division of the dataset into training (DS1) and testing (DS2) sets, adhering to the inter-patient paradigm outlined in Table 2 from Section 4.1. A segmentation size of 280 points is employed by the findings from Experiment 5.2.

Before segmentation, it is noteworthy that the ECG signals underwent a normalization process using the *z-score* technique. Normalization of a signal is an approach aimed at equalizing its levels to achieve uniformity. In this context, statistical parameters such as mean (μ) and standard deviation (σ) are used for calculating the *z-score*, forming part of the process[6]. The strategic choice of the *z-score* technique is driven by its ability to preserve the distribution of points within the ECG signals themselves, a feature that includes maintaining fiducial points. Given that ECG signals can exhibit a range of peaks with varying magnitudes, applying methods based on distance or neural networks can pose challenges, mainly due to issues related to gradient exploration [42].

Table 9 summarizes the performance of the GCN2 and GCN7 architectures with various groups of information aggregated using the VG method on the test set (DS2). For the GCN2 architecture, the RR and Stats groups exhibited the best outcomes, with average F_S scores of 68% and 73%, respectively. In the case of the GCN7 architecture,

the AvgII and Stats groups led the performance, achieving an average F_S of 77%.

Meanwhile, Table 10 presents the performance of the GCN2 and GCN7 architectures according to each information group aggregated through the VVG method. For GCN2, the RR and Stats groups again emerged as the top performers, with average F_S scores of 68% and 74%, respectively. Notably, the performance in this architecture is similar across both the VG and VVG methods. For GCN7, the top groups were StdII and Stats, with average F_S scores of 68% and 74%, respectively.

The Stats group, which contains the most aggregated information - primarily statistical data from ECG signals totaling 22 pieces of information - demonstrated the best performance in both architectures. Conversely, the smaller information groups showed variations in performance across the two architectures. The RR group aggregates five pieces of information, whereas the AvgII group comprises seven. Hence, more extensive and smaller quantities of aggregated information contributed similarly to enhancing the performance of the analyzed architectures.

The varying performance of GCNs across different information groups can be attributed to a combination of factors. These include the relevance of the aggregated information, the GCNs' capability to capture each group's distinctive features, the complexity of the GCN architectures, and the specific characteristics of the classified arrhythmias. An additional observation is that in the better-performing groups, the confusion matrices highlighted a tendency of the architectures to classify arrhythmic beats as normal less frequently

[6]$z = \frac{x - \mu}{\sigma}$, where μ is the mean and σ is the standard deviation of the ECG signal points.

Table 9: Summary of the performance of GCN architectures regarding aggregating information in the graphs using the VG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
II_V1	59.0	85.0	50.92	70.0	-	-	-	-	68.0	66.0	16.15	67.0	51.0	62.0	29.24	55.0	62.0
RR	74.0	94.0	28.63	83.0	52.0	2.0	0.52	5.0	77.0	91.0	14.25	83.0	**71.0**	**75.0**	**18.3**	**68.0**	75.0
DifII	73.0	94.0	31.07	82.0	41.0	2.0	0.79	4.0	77.0	87.0	13.63	81.0	68.0	74.0	19.28	67.0	74.0
AvgII	73.0	93.0	30.65	82.0	37.0	2.0	0.77	4.0	75.0	87.0	14.55	81.0	67.0	73.0	19.39	66.0	73.0
StdII	73.0	92.0	29.9	81.0	48.0	1.0	0.38	3.0	74.0	88.0	16.2	80.0	68.0	73.0	19.53	66.0	73.0
Stats	75.0	94.0	27.61	84.0	50.0	12.0	2.85	19.0	85.0	92.0	8.32	88.0	**74.0**	**77.0**	**16.26**	**73.0**	77.0
GCN7																	
II_V1	62.0	85.0	46.31	72.0	31.0	5.0	2.89	9.0	82.0	78.0	8.51	80.0	63.0	67.0	25.06	62.0	67.0
RR	75.0	90.0	26.72	81.0	62.0	30.0	4.44	41.0	88.0	89.0	6.31	88.0	77.0	78.0	15.47	76.0	78.0
DifII	76.0	88.0	24.07	82.0	60.0	29.0	4.68	39.0	86.0	93.0	7.73	89.0	76.0	78.0	14.76	76.0	78.0
AvgII	74.0	92.0	27.9	82.0	58.0	32.0	5.7	41.0	94.0	86.0	2.89	90.0	**78.0**	**79.0**	**14.68**	**77.0**	79.0
StdII	76.0	89.0	25.05	82.0	58.0	30.0	5.39	40.0	89.0	91.0	5.75	90.0	77.0	78.0	15.1	76.0	78.0
Stats	75.0	92.0	26.2	83.0	62.0	27.0	4.02	38.0	92.0	93.0	4.42	92.0	**78.0**	**80.0**	**14.5**	**77.0**	80.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

Table 10: Summary of the performance of GCN architectures regarding aggregating information in the graphs using the VVG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
II_V1	63.0	83.0	42.97	72.0	-	-	-	-	57.0	64.0	24.97	60.0	49.0	61.0	28.53	54.0	61.0
RR	75.0	92.0	26.91	82.0	52.0	2.0	0.39	3.0	74.0	92.0	16.33	82.0	**70.0**	**75.0**	**18.18**	**68.0**	75.0
DifII	75.0	92.0	26.83	82.0	42.0	3.0	1.07	6.0	74.0	91.0	16.07	82.0	68.0	74.0	18.19	67.0	74.0
AvgII	74.0	92.0	27.53	82.0	29.0	1.0	0.8	3.0	74.0	91.0	16.37	81.0	66.0	74.0	18.56	67.0	74.0
StdII	75.0	91.0	27.17	82.0	27.0	3.0	1.94	5.0	74.0	89.0	16.17	81.0	65.0	73.0	18.54	67.0	73.0
Stats	76.0	93.0	25.98	84.0	48.0	18.0	4.63	26.0	86.0	90.0	7.51	88.0	**74.0**	**77.0**	**15.57**	**74.0**	77.0
GCN7																	
II_V1	71.0	85.0	30.59	77.0	29.0	2.0	1.27	4.0	65.0	82.0	22.51	73.0	61.0	68.0	22.16	61.0	68.0
RR	75.0	91.0	25.92	82.0	18.0	8.0	9.17	11.0	76.0	77.0	12.86	76.0	64.0	70.0	18.24	67.0	70.0
DifII	76.0	90.0	25.41	82.0	21.0	12.0	11.49	15.0	76.0	73.0	11.79	75.0	65.0	69.0	18.09	67.0	69.0
AvgII	74.0	93.0	28.32	83.0	21.0	8.0	7.76	12.0	78.0	77.0	11.37	77.0	65.0	70.0	18.58	67.0	71.0
StdII	76.0	92.0	25.69	83.0	21.0	12.0	11.12	16.0	79.0	74.0	10.19	77.0	**66.0**	**71.0**	**17.6**	**68.0**	71.0
Stats	76.0	87.0	24.52	81.0	53.0	27.0	5.81	36.0	85.0	91.0	8.43	88.0	**74.0**	**77.0**	**15.43**	**74.0**	77.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

compared with the other groups. This suggests a more accurate detection of arrhythmias, a crucial aspect of practical ECG analysis.

5.4 Experiment 4: Swapping training and test sets

In Experiment 4, the MIT-BIH dataset is divided into a training set (DS1) and a testing set (DS2) following the inter-patient paradigm (see Table 3). It is observed that DS2 has a higher number of class S heartbeats compared to DS1. Given that class S presented challenges in performance with the analyzed architectures, this experiment examines whether reversing the datasets—using DS2 for training and DS1 for testing—can enhance the performance of GCN and CNN architectures in classifying class S.

The dataset configurations and training settings are consistent with those used in Experiment 5.3. For CNN training, a validation set comprising 10% of DS2 is used. In the case of GCNs, only the training (DS2) and testing (DS1) sets are used. The GCN architectures incorporated the two most effective information groups identified in Experiment 5.3 for each architecture under VG and VVG conversion methods.

Table 11 presents the performance of the architectures in terms of the dataset reversal between DS1 and DS2 using the VG method. The reversal

notably improved the performance in class S for both architectures, with a more significant increase observed in the sensitivity (Se) and F_S metrics, especially in the GCN7 architecture. This enhancement in class S performance positively impacted the overall performance of the architectures, leading to an increase in the average overall F_S, ranging from 75% to 84%. This suggests that training with a dataset with a higher representation of challenging classes like class S can result in better learning and generalization capabilities for the models.

The GCN7 architecture with the AvgII information group exhibited the best overall performance, achieving an 84% average F_S score. Comparing this result with the data from Table 9, the reversal of the datasets led to a performance increase of 9.09%. Similarly, for the CNN-2Conv architecture, there is an improvement in class S performance and overall effectiveness. The sensitivity (Se) and F_S for class S increased by 153.33% and 193.75%, respectively, while the general performance saw a 20.31% rise in the F_S metric.

Evaluating the performance of GCN architectures with the reversal of DS1 and DS2 datasets using the VVG method, results from Table 12 indicate that although there is not a significant increase in class S performance, the overall F_S scores improved compared to the results in Table 10. In the GCN2 architecture, the RR and Stats information groups showed an improvement of 11.76% and 5.4%, respectively, while in the GCN7 architecture, the StdII and Stats groups exhibited increases of 5.88% and 8.10%, respectively. Comparing VG and VVG methods across Tables 11 and 12, the VG method outperformed the VVG method. Therefore, using two leads to convert ECG signals into graphs, within the context of this research and with MIT-BIH DB, did not contribute as significantly as using a single lead to enhance the architectures' performance in the dataset reversal scenario. This finding suggests that, in this context, the GCN architectures studied cannot take advantage of an extra lead.

5.5 Experiment 5: Intra-patient Paradigm

In Experiment 5, the intra-patient paradigm is explored in contrast to the inter-patient approach. Unlike the inter-patient paradigm, where there is no overlap of data from the same patient between training and testing sets, the intra-patient approach allows for the presence of heartbeat data from the same patient in both training and testing datasets. This experiment, therefore, serves as a comparative analysis of the architecture performance across these two paradigms.

The dataset configurations and training settings follow those used in Experiment 5.3, with the distinction that the training and testing sets are randomly determined, allowing for the possibility of the same patient's data appearing in both sets.

Analyzing the results in Table 13 concerning the GCNs and VG method, the GCN2 architecture showed mixed performance changes with different information groups. The RR group showed a performance decline of 7.35% in the F_S metric compared with the inter-patient paradigm performance (Table 9), whereas the Stats group exhibited a performance increase of 2.74%. In the GCN7 architecture, the AvgII and Stats information groups demonstrated improvements of 2.6% and 12.98% in the F_S metric, respectively.

When comparing the two paradigms in the CNN-2Conv architecture, a substantial performance increase is observed in both class S and overall performance. The overall performance increase for the CNN-2Conv, measured by the F_S metric, is 50%. This significant improvement indicates that the intra-patient paradigm, where data from the same patient can appear in training and testing sets, may lead to better learning and adaptation of the models to the specific characteristics of individual patients' ECG signals.

Regarding the VVG method as shown in Table 14, the GCN2 architecture experienced a general performance decrease in both the RR and Stats information groups, with declines of 2.94% and 1.35%, respectively, compared with the results in Table 10. Conversely, in the GCN7 architecture, the StdII and Stats information groups showed improved performance with increases of 2.94% and 16.21% in the F_S metric, respectively.

This variation in performance between the GCN2 and GCN7 architectures under the intra-patient paradigm using the VVG method suggests that different information groups and GCN architectures respond uniquely to the challenges posed by this paradigm. Specifically, while some architectures and information groups may struggle to adapt to the overlapping patient data in the intra-patient setting, others, such as GCN7 with the

Table 11: Summary of the performance of GCN and CNN architectures regarding the datasets DS1 and DS2 reversal using the VG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
RR	82.0	85.0	18.66	83.0	27.0	27.0	8.33	27.0	79.0	74.0	13.57	77.0	75.0	75.0	15.54	75.0	75.0
Stats	89.0	84.0	10.38	86.0	39.0	50.0	8.72	44.0	90.0	90.0	7.18	90.0	84.0	83.0	8.91	83.0	83.0
GCN7																	
AvgII	86.0	90.0	14.22	88.0	63.0	57.0	3.74	60.0	85.0	83.0	9.64	84.0	83.0	84.0	11.3	84.0	84.0
Stats	86.0	84.0	12.7	85.0	55.0	51.0	4.68	53.0	83.0	88.0	12.52	85.0	82.0	82.0	11.81	82.0	82.0
CNN-2Conv																	
No reversal	64.0	67.0	33.28	65.0	18.0	15.0	16.54	16.0	88.0	90.0	6.44	89.0	63.0	65.0	20.92	64.0	65.0
Reversal	86.0	73.0	11.45	79.0	62.0	38.0	2.57	47.0	74.0	95.0	23.3	83.0	79.0	78.0	15.37	77.0	78.0

*Acc ⇒ Accuracy, Se ⇒ Sensitivity, +P ⇒ Positive Prediction, FPR ⇒ False Positive Rate, F$_s$ ⇒ F1-score.

Note: Values in bold indicate the best performance.

Table 12: Summary of the performance of GCN and CNN architectures regarding the datasets DS1 and DS2 reversal using the VVG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
RR	87.0	91.0	13.1	89.0	9.0	2.0	2.71	4.0	74.0	84.0	19.92	79.0	74.0	84.0	14.82	76.0	79.0
Stats	82.0	81.0	16.76	82.0	20.0	21.0	9.35	20.0	88.0	89.0	7.96	89.0	79.0	78.0	12.43	78.0	78.0
GCN7																	
StdII	81.0	77.0	17.1	79.0	14.0	17.0	12.51	15.0	77.0	77.0	15.54	77.0	73.0	71.0	16.0	72.0	71.0
Stats	81.0	88.0	19.55	84.0	36.0	35.0	7.1	35.0	90.0	82.0	6.51	85.0	80.0	80.0	12.99	80.0	80.0

*Acc ⇒ Accuracy, Se ⇒ Sensitivity, +P ⇒ Positive Prediction, FPR ⇒ False Positive Rate, F$_s$ ⇒ F1-score.

Note: Values in bold indicate the best performance.

Table 13: Summary of the performance of GCN and CNN architectures regarding the intra-patient paradigm using the VG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
RR	76.0	87.0	24.64	81.0	40.0	-	0.08	-	63.0	85.0	25.8	73.0	64.0	70.0	20.27	63.0	70.0
Stats	77.0	91.0	23.75	83.0	65.0	21.0	2.7	32.0	82.0	93.0	10.61	87.0	76.0	78.0	15.21	75.0	78.0
GCN7																	
AvgII	77.0	94.0	24.3	85.0	93.0	31.0	0.56	46.0	86.0	94.0	7.62	90.0	83.0	82.0	14.03	79.0	82.0
Stats	85.0	93.0	13.88	89.0	87.0	60.0	2.15	71.0	90.0	95.0	5.54	92.0	87.0	87.0	8.77	87.0	87.0
CNN-2Conv																	
Inter-patient	62.0	72.0	38.07	67.0	21.0	14.0	12.57	17.0	88.0	86.0	6.1	87.0	63.0	66.0	22.27	64.0	66.0
Intra-patient	95.0	97.0	4.11	96.0	94.0	90.0	1.4	92.0	99.0	98.0	0.69	98.0	96.0	96.0	2.42	96.0	96.0

*Acc ⇒ Accuracy, Se ⇒ Sensitivity, +P ⇒ Positive Prediction, FPR ⇒ False Positive Rate, F$_s$ ⇒ F1-score.

Note: Values in bold indicate the best performance.

StdII and Stats groups, appear to thrive, showing significant performance improvements. This observation highlights the importance of selecting the appropriate combination of architecture and information groups to optimize performance in a clinical or experimental setting.

Overall, the results underscore that the intra-patient paradigm generally yields better outcomes than the inter-patient paradigm in the performance of the examined architectures despite not representing a scenario closer to real-world conditions where data from a new patient are not used during model training.

Throughout the experiments, it is observed that the VVG method exhibits high computational complexity in terms of both time and space due to

Table 14: Summary of the performance of GCN and CNN architectures regarding the intra-patient paradigm using the VVG method.

Information	N				S				V				Weighted Average				Acc*
	+P*	Se*	FPR*	F$_s$*	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	+P	Se	FPR	F$_s$	
GCN2																	
RR	75.0	86.0	24.48	80.0	96.0	8.0	0.09	15.0	67.0	89.0	22.65	76.0	76.0	72.0	19.13	66.0	72.0
Stats	76.0	93.0	25.11	84.0	68.0	14.0	1.6	23.0	81.0	94.0	11.25	87.0	76.0	78.0	15.85	73.0	78.0
GCN7																	
StdII	75.0	87.0	25.45	81.0	81.0	18.0	1.06	30.0	70.0	85.0	18.75	77.0	74.0	73.0	18.45	70.0	73.0
Stats	87.0	91.0	11.67	89.0	80.0	62.0	3.81	69.0	88.0	94.0	6.61	91.0	86.0	86.0	8.43	86.0	86.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

the utilization of two leads for mapping ECG signals. More efficient data structures (organization of programming codes) address the time complexity, allowing for heartbeats of 280 points to be mapped in approximately 0.3 seconds. However, space complexity remains challenging in mapping ECG signals and processing the generated graphs during training and testing. Although data sub-sampling is initially applied to mitigate this issue, exploring other solutions that allow the full potential of graph convolutional networks to be leveraged is necessary. This exploration might involve investigating alternative data representations or optimizing graph processing algorithms to enhance the efficiency of space utilization while maintaining the integrity and effectiveness of the analysis.

5.6 Experiment 6: Comparison of the Proposed Method

In this final experiment, a comparative analysis is conducted between the method proposed in this work and the study by Garcia et al. [12], which employed a graph modeling and SVM classifier. The rationale for selecting this study for comparison stems from its status as a significant baseline reference in this work and its methodological parallels with the approach presented herein. Modifications have been made to the original study by [12] to enable direct comparison. Notably, this included a 10% subsampling of class N heartbeats; specifically, only the tenth beat in every sequence of ten is selected. Furthermore, the same datasets, DS1.1 and DS1.2, as detailed in Table 6, are employed for tuning the SVM model parameters. The comparison extended across three distinct scenarios:

- Garcia et al. (2017) [12]:

  - **Scenario 1**: Features extracted from the graph network based on complex networks;
  - **Scenario 2**: Features extracted from the graph network based on complex networks along with RR interval features;
  - **Scenario 3**: Features extracted from the graph network based on complex networks, RR interval features, and statistical characteristics of the graph network;

- **Proposed Method**:

  - **Scenario 1**: Experiment 3 of the GCN7 architecture with information from the II_V1 group and the VG method;
  - **Scenario 2**: Experiment 3 of the GCN7 architecture with information from the RR group and the VG method;
  - **Scenario 3**: Experiment 3 of the GCN7 architecture with information from the Stats group and the VG method;

The comparative results in Table 15 highlight the efficacy of our introduced method when contrasted with the findings of [12]. In both Scenario 1 and Scenario 2, the proposed method delivered improved results for minority classes, specifically the arrhythmic categories, but encountered challenges with class N, where it displayed a modest reduction in performance in Positive Prediction (+P) and F_s metrics. A similar trend is evident in Scenario 3, with class N exhibiting a decline in performance concerning the +P, Sensitivity (Se), and F_s metrics. However, it is essential to acknowledge that in Scenario 3, the statistical features used by [12] differ from the information used in the proposed method. Despite this disparity, a comparative analysis enables an approximate evaluation of the proposed scenario.

Table 15: Comparison of the proposed method with the work of Garcia et al. [12].

Scenarios	Work	N				S				V				Acc*
		+P*	Se*	FPR*	F_s*	+P	Se	FPR	F_s	+P	Se	FPR	F_s	
1	Garcia et al. [12]	**79.60**	74.2	60.83	**76.80**	0.40	0.20	7.90	0.30	39.40	71.9	15.13	50.90	65.20
	Proposed method	62.0	**85.0**	**46.31**	71.47	**31.0**	**5.0**	**2.89**	**8.61**	**82.0**	**78.0**	**8.51**	**79.95**	**67.0**
2	Garcia et al. [12]	**86.60**	87.40	43.36	**87.0**	21.10	2.80	**1.38**	4.90	46.80	83.90	13.05	60.10	77.0
	Proposed method	75.0	**90.0**	**26.72**	81.81	**62.0**	**30.0**	4.44	**40.43**	**88.0**	**89.0**	**6.31**	**88.50**	**78.0**
3	Garcia et al. [12]	**90.20**	**95.30**	48.90	**92.70**	55.30	17.70	**1.37**	26.80	68.20	78.40	**3.48**	72.90	**87.0**
	Proposed method	78.0	92.0	**26.20**	84.82	**62.0**	**27.0**	4.02	**37.62**	**92.0**	**93.0**	4.42	**92.49**	80.0

*$Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

Note: Values in bold indicate the best performance.

A key observation is that the proposed method demonstrates superior performance in the arrhythmic classes (S and V) compared with the normal class (N). This underscores its effectiveness in differentiating between normal and arrhythmic heartbeats, essential for minimizing false predictions. This aspect is of significant concern in clinical applications and impacts the method's reliability.

5.7 Experimental Decisions

Acknowledging the stochastic nature of neural network training, we conducted each experiment only once. This decision was guided by preliminary tests, wherein selected architectures were run ten times each, revealing minimal variation and deviation in the outcomes, as detailed in Table 16. These initial experiments, focusing on the GCN2 and GCN7 architectures with the Stats information set and employing the VG method, demonstrated significant consistency in standard deviation and variance across evaluated metrics, even with the introduction of randomness through varied training seeds.[7] Given the observed stability and the high computational cost of multiple runs, we concluded that a single execution would suffice, ensuring efficient resource use while maintaining confidence in the reliability and reproducibility of our results.

6 Conclusion

This study proposed a method for classifying arrhythmias in ECG signals by mapping them into graphs and classifying them using Graph Convolutional Networks (GCNs), following the inter-patient paradigm and AAMI standards. The central research question is whether graph representations of ECG signals, through VG and VVG methods, could enhance arrhythmia classification performance using GCNs.

The findings indicated that simpler GCN architectures yielded better results than more complex ones, suggesting that simplicity in GCN structures can more effectively capture essential data characteristics and avoid unnecessary noise. The results presented in Section 5.1 showed that GCN2 outperformed GCN240 in most metrics, obtaining 25.8%, 37.5%, 24.4%, and 37.5% in +P, Se, Fs, and Acc of increase, respectively.

Simpler architectures are computationally more efficient, a necessary factor in resource-constrained scenarios. Including extrinsic information in the graphs improved the VG and VVG methods, as the selected information accurately captured the ECG signal morphology. While both VG and VVG showed promise, VG is more efficient for the explored GCN architectures. When we observe, the results of Experiment 3 showed that VG exhibited better performances than VVG when comparing the weighted average metric means, with a more pronounced increase of 13.67%, 8.3%, 9.79%, and 8% for +P, Se, Fs, and Acc, respectively, in the GCN7 architecture. However, the challenge of classifying the S class remained, especially under the inter-patient paradigm, even when reversing the DS1 and DS2 data sets. Conversely, the intra-patient paradigm achieved better outcomes, although it does not fully reflect real-world scenarios. The results suggest that it is feasible to classify arrhythmias in ECG signals using GCNs with VG and VVG for signal graph mapping, with the advantage of requiring no preprocessing or noise removal from ECG signals. Still, there is room

---

[7]We introduced randomness via the **torch.manual_seed(random.randint(1,100000)** command, utilizing the *torch* and *random* libraries to generate a random number between 1 and 100000 for each execution.

Table 16: Results of experimental decisions.

#Execution	Acc*	+P*	Se*	FPR*	F_s^*	Time(s)
Experiment 1: GCN2_Stats						
#1	77.56	74.62	77.56	16.33	72.17	5408
#2	77.33	73.41	77.33	16.07	72.78	5435
#3	78.05	75.50	78.04	16.21	72.84	5528
#4	77.40	73.84	77.40	16.41	72.56	5521
#5	77.70	74.29	77.70	16.04	72.96	5528
#6	77.80	74.41	77.80	15.98	73.05	5555
#7	77.37	73.64	77.37	16.23	72.32	5567
#8	77.48	73.50	77.48	15.95	72.90	5578
#9	76.88	72.77	76.88	16.20	72.75	5598
#10	77.48	73.71	77.48	16.22	72.48	5617
Average	77.50	73.97	77.51	16.16	72.68	5533
Standard Deviation	0.003	0.007	0.003	0.14	0.003	63.53
Variance	8.8e-6	5.23e-5	8.8e-6	0.02	7.5e-6	4.04e3
Experiment 2: GCN7_Stats						
#1	78.15	76.33	78.15	15.17	75.52	8124
#2	79.42	78.32	79.42	15.16	76.95	8174
#3	79.56	78.20	79.58	14.30	77.74	8290
#4	80.06	79.66	80.06	15.0	77.26	8337
#5	78.86	77.21	78.86	14.50	76.83	8390
#6	79.75	78.21	79.75	14.09	77.56	8370
#7	79.04	77.66	79.04	15.17	76.37	8378
#8	78.62	77.00	78.62	15.19	75.08	8395
#9	78.37	77.33	78.37	15.25	77.76	8462
#10	79.64	78.49	79.64	14.65	77.56	8437
Average	79.15	77.84	79.15	14.84	76.66	8335
Standard Deviation	0.006	0.009	0.006	0.405	0.009	104.26
Variance	3.66e-5	7.90e-5	3.66e-5	0.16	8.0e-5	1.09e4

* $Acc \Rightarrow$ Accuracy, $Se \Rightarrow$ Sensitivity, $+P \Rightarrow$ Positive Prediction, $FPR \Rightarrow$ False Positive Rate, $F_s \Rightarrow$ F1-score.

for improvement and further research to harness GCNs' potential fully.

A significant limitation encountered in developing this method is the computational complexity of the VVG method in mapping ECG signals and training the GCNs, which is related to processing time and required space. More efficient data structures are initially used to reduce the ECG signal mapping time. For space constraints, data balancing is applied to lessen the number of generated graphs, allowing experiments on a data subset. However, further alternatives could be explored to address this challenge.

References

[1] Cohen, A.: Biomedical signal processing: Compression and automatic recognition **2** (2019)

[2] Pandey, S.K., Janghel, R.R.: Automatic detection of arrhythmia from imbalanced ecg database using cnn model with smote. Australasian physical & engineering sciences in medicine **42**(4), 1129–1139 (2019)

[3] Cheng, P., Dong, X.: Life-threatening ventricular arrhythmia detection with personalized features. IEEE access **5**, 14195–14203 (2017)

[4] Luz, E.J.d.S., Schwartz, W.R., Cámara-Chávez, G., Menotti, D.: Ecg-based heartbeat classification for arrhythmia detection: A survey. Computer methods and programs in biomedicine **127**, 144–164 (2016)

[5] Zaorálek, L., Platoš, J., Snášel, V.: Patient-adapted and inter-patient ecg classification using neural network and gradient boosting. Neural Network World **28**(3), 241–254 (2018)

[6] Hannun, A.Y., Rajpurkar, P., Haghpanahi, M., Tison, G.H., Bourn, C., Turakhia, M.P., Ng, A.Y.: Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature medicine **25**(1), 65 (2019)

[7] ANSI/AAMI: Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms. American National Standards Institute, Inc. (ANSI), Association for the Advancement of Medical Instrumentation (AAMI). ANSI/AAMI/ISO EC57, 1998-(R)2008 (2008)

[8] De Chazal, P., O'Dwyer, M., Reilly, R.B.: Automatic classification of heartbeats using ecg morphology and heartbeat interval features. IEEE transactions on biomedical engineering **51**(7), 1196–1206 (2004)

[9] Luz, E., Menotti, D.: How the choice of samples for building arrhythmia classifiers impact their performances. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 4988–4991

(2011). IEEE

[10] Mousavi, S., Afghah, F.: Inter-and intra-patient ecg heartbeat classification for arrhythmia detection: a sequence to sequence deep learning approach. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1308–1312 (2019). IEEE

[11] Essa, E., Xie, X.: An ensemble of deep learning-based multi-model for ecg heartbeats arrhythmia classification. IEEE Access **9**, 103452–103464 (2021)

[12] Garcia, G., Moreira, G., Menotti, D., Luz, E.: Inter-patient ecg heartbeat classification with temporal vcg optimized by pso. Scientific reports **7**(1), 1–11 (2017)

[13] Barabási, A.-L.: Network science. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **371**(1987), 20120375 (2013)

[14] Ren, W., Jin, N.: Vector visibility graph from multivariate time series: a new method for characterizing nonlinear dynamic behavior in two-phase flow. Nonlinear Dynamics **97**, 2547–2556 (2019)

[15] Zhang, J., Small, M.: Complex network from pseudoperiodic time series: Topology versus dynamics. Physical review letters **96**(23), 238701 (2006)

[16] Sun, X., Small, M., Zhao, Y., Xue, X.: Characterizing system dynamics with a weighted and directed network constructed from time series data. Chaos: An Interdisciplinary Journal of Nonlinear Science **24**(2), 024402 (2014)

[17] Donner, R.V., Zou, Y., Donges, J.F., Marwan, N., Kurths, J.: Recurrence networks—a novel paradigm for nonlinear time series analysis. New Journal of Physics **12**(3), 033025 (2010)

[18] Donges, J.F., Donner, R.V., Trauth, M.H., Marwan, N., Schellnhuber, H.-J., Kurths, J.: Nonlinear detection of paleoclimate-variability transitions possibly related to human evolution. Proceedings of the National Academy of

Sciences **108**(51), 20422–20427 (2011)

[19] Lacasa, L., Luque, B., Ballesteros, F., Luque, J., Nuno, J.C.: From time series to complex networks: The visibility graph. Proceedings of the National Academy of Sciences **105**(13), 4972–4975 (2008)

[20] Luque, B., Lacasa, L., Ballesteros, F., Luque, J.: Horizontal visibility graphs: Exact results for random time series. Physical Review E **80**(4), 046103 (2009)

[21] Gotoda, H., Kinugawa, H., Tsujimoto, R., Domen, S., Okuno, Y.: Characterization of combustion dynamics, detection, and prevention of an unstable combustion state based on a complex-network theory. Physical Review Applied **7**(4), 044027 (2017)

[22] Freitas, V.L.S., Lacerda, J.C., Macau, E.E.N.: Complex Networks Approach for Dynamical Characterization of Nonlinear Systems. International Journal of Bifurcation and Chaos **29**(13), 1950188–512 (2019) https://doi.org/10.1142/S0218127419501888

[23] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M.: Graph neural networks: A review of methods and applications. AI Open **1**, 57–81 (2020)

[24] Sperduti, A., Starita, A.: Supervised neural networks for the classification of structures. IEEE Transactions on Neural Networks **8**(3), 714–735 (1997)

[25] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE transactions on neural networks **20**(1), 61–80 (2008)

[26] Micheli, A.: Neural network for graphs: A contextual constructive approach. IEEE Transactions on Neural Networks **20**(3), 498–511 (2009)

[27] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)

[28] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems **32**(1), 4–24 (2020)

[29] Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A., Vandergheynst, P.: The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. IEEE signal processing magazine **30**(3), 83–98 (2013)

[30] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv:1609.02907 (2016)

[31] Hamilton, W.L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 1025–1035 (2017)

[32] Cao, M., Zhao, T., Li, Y., Zhang, W., Benharash, P., Ramezani, R.: Ecg heartbeat classification using deep transfer learning with convolutional neural network and stft technique. In: Journal of Physics: Conference Series, vol. 2547, p. 012031 (2023). IOP Publishing

[33] Gai, N.D.: Ecg beat classification using machine learning and pre-trained convolutional neural networks. arXiv preprint arXiv:2207.06408 (2022)

[34] Mathews, S.M., Kambhamettu, C., Barner, K.E.: A novel application of deep learning for single-lead ecg classification. Computers in biology and medicine **99**, 53–62 (2018)

[35] Zhao, X., Liu, Z., Han, L., Peng, S.: Ecgnn: Enhancing abnormal recognition in 12-lead ecg with graph neural network. In: 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 1411–1416 (2022). IEEE

[36] Duong, L.T., Doan, T.T.H., Chu, C.Q., Nguyen, P.T.: Fusion of edge detection and graph neural networks to classifying electrocardiogram signals. Expert Systems with

Applications **225**, 120107 (2023) https://doi.org/10.1016/j.eswa.2023.120107

[37] Moody, G.B., Mark, R.G.: The mit-bih arrhythmia database on cd-rom and software for use with it. In: [1990] Proceedings Computers in Cardiology, pp. 185–188 (1990). IEEE

[38] Moody, G.B., Mark, R.G.: The impact of the mit-bih arrhythmia database. IEEE Engineering in Medicine and Biology Magazine **20**(3), 45–50 (2001)

[39] Sraitih, M., Jabrane, Y., Atlas, A.: An overview on intra-and inter-patient paradigm for ecg heartbeat arrhythmia classification. In: 2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA), pp. 1–7 (2021). IEEE

[40] Kojima, R., Ishida, S., Ohta, M., Iwata, H., Honma, T., Okuno, Y.: kgcn: a graph-based deep learning framework for chemical structures. Journal of Cheminformatics **12**(1), 1–10 (2020)

[41] Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? arXiv preprint arXiv:1810.00826 (2018)

[42] Shobanadevi, A., Veeramakali, T.: Classification and interpretation of ecg arrhythmia through deep learning techniques (2023)