# Automatic Cardiac Arrhythmia Classification Using Residual Network Combined With Long Short-Term Memory

Yun Kwan Kim, Minji Lee, Hee Seok Song, and Seong-Whan Lee, *Fellow, IEEE*

***Abstract*— Diagnosis and classification of arrhythmia, which is associated with abnormal electrical activities in the heart, are critical for clinical treatments. Previous studies focused on the diagnosis of atrial fibrillation, which is the most common arrhythmia in adults. The classification performance achieved by studies on other arrhythmia types is not satisfactory for clinical use owing to the small number of classes (minority classes). In this study, we propose a novel framework for automatic classification that combines a residual network with a squeeze-and-excitation block and a bidirectional long short-term memory. Eight-, four-, and two-class performances were evaluated on the MIT-BIH arrhythmia database (MITDB), the MIT-BIH atrial fibrillation database (AFDB), and the PhysioNet/Computing in the cardiology challenge 2017 database (CinC DB), respectively, and they were superior to the performance achieved by conventional methods. In addition, the classwise F1-score in the minority classes was higher than those of the methods adopted in existing studies. To measure the generalization ability of the proposed framework, AFDB and CinC DB were tested using an MITDB-trained model, and superior performance was achieved compared with ShallowConvNet and DeepConvNet. We performed a cross-subject experiment using AFDB and obtained a statistically higher performance using the proposed method compared with typical machine learning methods. The proposed framework can enable the direct diagnosis of arrhythmia types in clinical trials based on the accurate detection of the minority class.**

***Index Terms*— Arrhythmia classification, augmentation, electrocardiography (ECG), few shot, long short-term memory, residual network (ResNet), squeeze-and-excitation (SE) block.**

## I. Introduction

CARDIAC arrhythmia is a condition in which the heart rate is irregular, either extremely fast or extremely slow [1]. Although most arrhythmias are not serious, some can result in stroke, heart failure, and even sudden death [2]. The incidence of arrhythmias has increased in recent years. Arrhythmias incidences have increased in recent years. In particular, the frequency of cardiac rhythm abnormalities in middle-aged to older community-dwelling adults is substantial [3]. Moreover, it is important to accurately diagnose various types of arrhythmias to reduce the associated threats.

Arrhythmias are broadly divided into three types: supraventricular, ventricular, and bradycardia [4]. Supraventricular arrhythmia includes atrial fibrillation (AFIB), atrial flutter (AFL), atrioventricular junctional rhythm (AVR), and supraventricular tachycardia (SVT). In AFIB, p waves are not visible for fast heart signals and low-amplitude levels. AFL is completely similar to AFIB; however, the electrical impulses of AFL are organized, whereas those in AFIB are not. In AVR, electrical activation starts near or within the atrioventricular node, instead of the sinoatrial node [4]. SVT occurs when the heart rate is high in the upper heart chambers [5]. Ventricular tachycardia (VT), which is a series of more than three consecutive abnormal heartbeats at a rate of more than 100 beats per minute, is a typical ventricular arrhythmia [5]. Bradycardia includes sinus bradycardia (SBR) with a lower than the normal heart rate [6]. Normal sinus rhythm (NSR) is defined as the rhythm of a healthy heart. Eight types of arrhythmia classifications, including AFIB, AFL, AVR, SVT, VT, SBR, NSR, and Other, are typical categorizations based on the location of occurrence, which can be considerably helpful when clinicians plan diagnosis and treatment [7].

Electrocardiography (ECG) is a noninvasive diagnostic tool that evaluates changes in the electrical activity of the heart over time by graphically recording its rhythm and electrical activity [8], enabling the detection of abnormal heart-related disorders. Holter recordings require an accurate diagnosis of abnormal heart rhythms and beats for 24 h or more to detect arrhythmia. However, the ECG signal is nonlinear with a low amplitude, leading cardiologists to neglect small changes [9], [10]. In addition, the manual diagnosis of ECG signals is time-consuming and cumbersome because it is recorded over long periods. To overcome these problems, many automatic algorithms have been developed to improve the accuracy of ECG diagnosis.

Many studies on arrhythmia classification have focused on classifying AFIB and NSR [11] because they are the most common arrhythmia types in adults [12]. However, many other

Manuscript received 30 December 2021; revised 22 April 2022; accepted 20 May 2022. Date of publication 10 June 2022; date of current version 30 June 2022. This work was supported by the Korean Medical Device Development (KMDF) Grant funded by the Korean Government under Grant KMDF-PR-20200901-0173. The Associate Editor coordinating the review process was Huang-Chen Lee. *(Corresponding authors: Hee Seok Song; Seong-Whan Lee.)*

Yun Kwan Kim is with the Department of Brain and Cognitive Engineering, Korea University, Seoul 02841, Republic of Korea, and also with Technology Development, Seers Technology Company Ltd., Seongnam-si 13558, Republic of Korea (e-mail: ykwin@korea.ac.kr).

Minji Lee is with the Department of Brain and Cognitive Engineering, Korea University, Seoul 02841, Republic of Korea (e-mail: minjilee@korea.ac.kr).

Hee Seok Song is with Technology Development, Seers Technology Company Ltd., Seongnam-si 13558, Republic of Korea (e-mail: sam.song@seerstech.com).

Seong-Whan Lee is with the Department of Artificial Intelligence, Korea University, Seoul 02841, Republic of Korea (e-mail: sw.lee@korea.ac.kr).

Digital Object Identifier 10.1109/TIM.2022.3181276

types of arrhythmias have recently been reported [3]. It is important to accurately diagnose various types of arrhythmias because treatment methods vary depending on the type. Among the diverse arrhythmia conditions, AFIB and AFL have the same patterns, and highly trained cardiologists are unable to accurately determine the correct arrhythmia types from ECG signals. Rajpurkar *et al.* [13] studied the performance of cardiologists in classifying arrhythmias and found that the average classification performance for AFIB, AFL, and VT was less than 70%. Cardiologists have different views on distinguishing between arrhythmias. Therefore, accurate automated arrhythmia detection systems for various types of arrhythmias are required in clinical settings.

Recently, deep learning has been actively used for arrhythmia diagnosis [25], resulting in significant performance improvements. However, challenges remain in their practical application. First, although ECG features extracted using deep learning algorithms may provide useful information for the automatic identification of cardiac arrhythmias [26], it is difficult to extract the unique features of arrhythmias. A single arrhythmia may have diverse ECG morphologies in different patients because ECG signals have distinctive morphological and temporal features. Therefore, it is challenging to extract unique ECG features for certain types of cardiac diseases using deep learning. Second, many studies have proposed arrhythmia classification methods based on a skewed class distribution consisting of majority and minority classes. The majority class refers to classes with abundant examples in the entire dataset, and the minority class refers to those with few classes within the group [27]. Moreover, when a majority class occupies most of the classes in a database, the minority class has a limited representation [28]. Imbalanced classification models have low predictive accuracy for minority classes because most deep learning algorithms used for classification are designed assuming an equal number of examples for each class [29].

In this study, we aimed at addressing the aforementioned challenges by developing a reliable framework for the fully automated classification of a large number of arrhythmias (AFIB, AFL, AVR, SVT, VT, SBR, NSR, and Other). This is achieved by combining the residual network (ResNet) [30] with a squeeze-and-excitation (SE) block [31] and bidirectional long short-term memory (biLSTM) [32]. An advantage of the proposed framework is the extraction of distinctive features using ResNet by attaching an SE block and biLSTM. In addition, we used the synthetic minority oversampling technique (SMOTE) as a data augmentation method proposed in 2002 to solve the class imbalance problem [33]. The implementation of SMOTE can now be found in many software libraries. The algorithm selects examples close to a feature space, draws lines between the examples in that space, and draws new samples at points along that line. We used the MIT-BIH arrhythmia database (MITDB) [34], the MIT-BIH atrial fibrillation database (AFDB) [34], and the PhysioNet/Computing in the cardiology challenge 2017 database (CinC DB) [35]. In addition, we measured the generalization ability of the proposed method, which is crucial because deep learning models sometimes do not perform well for new datasets despite their typically reasonable performance [36], [37]. In particular, we employed the few-shot learning method using independent datasets, which refers to teaching models as a task with a small number of annotated examples [38], [39]. The few-shot learning method is one of the methods used to measure generalization ability [40], [41]. We first trained the model using MITDB and evaluated it using AFDB and CinC DB. We hypothesized that the generalization performance of the proposed model would be higher than that of other models, such as ShallowConvNet [42] and DeepConvNet [43]. We additionally performed a cross-subject experiment as a leave-one-subject-out approach to measure the generalization ability [44]. In the cross-subject experiment, one subject was chosen as a test subject, while the training models were trained on the rest of the subjects in the same database. We performed only the AFDB because there was no common arrhythmia among the subjects in MITDB and CinC DB. Two classes, AFIB and NSR, were found to be in common among the 21 subjects in the AFDB. In this regard, a cross-subject experiment was performed using 21 subjects from the AFDB, and other typical machine learning methods were compared.

The main contributions of this study can be summarized as follows.

1) Our proposed framework combines ResNet with SE block and biLSTM to extract features from raw ECG data to obtain unique intersubject characteristics.
2) We have demonstrated that combining the proposed method with SMOTE showed an effective framework for solving imbalance problems in the arrhythmia classification compared to other augmentation methods.
3) We have demonstrated that our proposed method exhibits the highest generalization ability compared to other models that are widely used in related areas.

## II. Related Works

Many deep learning frameworks have recently been proposed using ECG signals for arrhythmia classification. In addition, various data augmentation strategies have been used to solve the class imbalance problem and achieve a higher classification performance. Table I shows various performance metrics presented in the existing studies to help explore and compare the performance of different methods for classifying cardiac arrhythmia.

### A. Data Augmentation for Class Imbalance

Several data augmentation methods have been used to increase the number of minority classes. Indeed, it helps to intuitively impose nonuniform misclassification costs by changing the class distribution of the training data to uniformly change the highly skewed dataset [45]. However, they do not affect the absolute rarity of both rare and rare cases [46]. In addition, these approaches can cause overfitting [47]. Therefore, it is important to use an appropriate data augmentation method depending on the problem.

TABLE I
SUMMARY OF RELATED WORKS ON THE CLASSIFICATION OF ARRHYTHMIA

| Author | Year | Database | No. of Class | ECG Class | Augmentation Method | Performance (%) |
|---|---|---|---|---|---|---|
| **Andreotti et al. [14]** | 2017 | CinC DB | 4 | AFIB, Noise, NSR, Other | - | F1-score: 83.00 |
| **Acharya et al. [9]** | 2017 | MITDB + AFDB + CUDB | 4 | AFIB, AFL, NSR, VT | - | Acc: 94.00<br>Sensitivity: 99.00<br>Specificity: 81.00<br>F1-score: 71.54 |
| **Sujadevi et al. [15]** | 2017 | AFDB + NSRDB | 2 | AFIB, NSR | - | F1-score: 100.00 |
| **Pławiak et al. [16]** | 2018 | MITDB | 17 | AFIB, AFL, AVR, SVT, NSR, WPW, PAC, PVC, Ventricular B, Ventricular T, VT, VFL, F, LBBB, RBBB, II-AVB, Pacemaker rhythm | - | Acc: 91.00<br>F1-score: 81.49 |
| **Yıldırım et al. [17]** | 2018 | MITDB | 17 | AFIB, AFL, AVR, SVT, NSR, WPW, PAC, PVC, Ventricular B, Ventricular T, VT, VFL, F, LBBB, RBBB, II-AVB, Pacemaker rhythm | - | Acc: 91.33<br>Specificity: 99.41<br>Precision: 89.52<br>Recall: 83.91<br>F1-score: 85.1 |
| **Faust et al. [18]** | 2018 | AFDB | 2 | AFIB, NSR | - | F1-score: 99.77 |
| **Chen et al. [19]** | 2020 | MITDB | 6 | AFIB, AFL, SBR, NSR, B, P | - | Acc: 99.32<br>F1-score: 90.82 |
| **Liang et al. [20]** | 2020 | CPSC | 9 | AFIB, I-AVB, LBBB, RBBB, PAC, PVC, STD, STE, NSR | ROS | Sensitivity 74.30<br>Specificity: 97.50<br>F1-score: 80.00 |
| **He et al. [21]** | 2020 | CPSC | 9 | AFIB, I-AVB, LBBB, RBBB, PAC, PVC, STD, STE, NSR | ROS | F1-score: 80.60 |
| **Yildirim et al. [22]** | 2020 | CU + SPH | 7 | AFIB, AFL, SI, SBR, NSR, ST, SVT | - | Acc: 92.24 |
| **Shaker et al. [23]** | 2020 | MITDB | 15 | NOR, LBBB, RBBB, PAC, PVC, AP, VF, VFN, BAP, NE, FPN, VE, NP, AE, UN | GAN | Acc: 98.30<br>Specificity: 99.47<br>Precision: 90.00<br>Recall: 99.77<br>F1-score: 85.1 |
| **Murat et al. [24]** | 2021 | CU + SPH | 4 | AFIB, GSVT, SBR, NSR | - | Acc: 90.30 |

MIT-BIH arrhythmia database, MITDB. MIT-BIH atrial fibrillation database, AFDB. Creighton university ventricular tachyarrhythmia database, CUDB. PhysioNet/Computing in the cardiology challenge 2017 database, CinC DB. China's physiological signal challenge, CPSC Chapman University, CU. Shaoxing People's Hospital (SPH). Atrial fibrillation, AFIB. Atrial flutter, AFL. Normal sinus rhythm, NSR. Atrioventricular junctional rhythm, AVR. Suparventricular tachycardia. SVT. Pre-excitation, WPW. Ventricular tachycardia, VT. Ventricular flutter, VFL. Second-degree atrioventricular block, II-AVB. Random oversampling, ROS. Bigeminy, B. Pacing rhythm, P. first-degree atrioventricular block, I-AVB. left bundle branch block, LBBB. right bundle branch block, RBBB. premature atrial contraction, PAC. premature ventricular contraction, PVC. ST-segment depression, STD. ST-segment elevation, STE. Normal beat, NOR. Aberrated atrial premature, AP. Ventricular flutter wave, VF. Fusion of ventricular and normal, F. Blocked atrial premature, BAP. Nodal escape, NE. Fusion of paced and normal, FPN. Ventricular escape, VE. Nodal premature, NP. Atrial escape, AE. Unclassifiable, UN. Sinus irregularity, SI. Accuracy, Acc.

He *et al.* [21] used random oversampling (ROS) to improve arrhythmia classification performance by generating a uniform data distribution of arrhythmia classes using the china physiological signal challenge database. There are nine types of arrhythmias: NSR, AFIB, first-degree atrioventricular block, left bundle branch block, right bundle branch block, premature ventricular contraction, premature atrial contraction, ST-segment depression, and ST-segment elevation. They achieved F1-score close to or higher than 78.0% for all arrhythmia types. However, a comparison of the results with and without ROS was not presented. Liang *et al.* [20] used ROS to classify nine types of arrhythmias in the same database. Although the overall sensitivity, specificity, and F1-score were 74.3%, 97.5%, and 80.0%, respectively, the enhancing effect was not compared. Shaker *et al.* [23] proposed a generative adversarial network (GAN) to restore the balance of MITDB for 15 classes of heartbeats in MITDB. They introduced an end-to-end approach based on a deep convolutional neural network (CNN). They obtained an overall accuracy, precision, specificity, and sensitivity of 98.30%, 90.00%, 99.77%, and 99.23%, respectively.

ROS, which has the advantage of rebalancing the class distribution [47], has several limitations. It is prone to overfitting because learning algorithms tend to focus on duplicated examinations of minority classes [33]. In addition, because the sampling process is random, it becomes difficult for the decision function to find a clear borderline among the classes [48]. Therefore, other data augmentation methods, such as the GAN and the adaptive synthetic sampling approach (ADASYN), have recently been used. A GAN utilizes two deep learning networks to produce synthetic data based on the potential distribution of real data samples [49]. Training this network is difficult, and generating results is often considered to be complex [50]. ADASYN is an extension of the SMOTE, which uses a weighted distribution for different minority classes [51]. SMOTE outperforms ADASYN when the degree of class imbalance increases [52]. Therefore, we selected SMOTE to compensate for the class imbalance.

### *B. Deep Learning for Classification*

Andreotti *et al.* [14] compared the performance of a feature-based classifier with that of a CNN for four classes: NSR, AFIB, noise, and other rhythms. They used the CinC DB and showed that the CNN obtained a higher F1-score (83.0%) than the feature-based classifier (79.0%). Moreover, for the

minority class, the F1-score for AFIB and other rhythms were 78.0% and 78.0%, respectively. Acharya *et al.* [9] presented an 11-layer deep CNN for automatically detecting ECG segments. They considered AFIB, AFL, VF, and NSR classes and used the MITDB, AFDB, and the Creighton University ventricular tachyarrhythmia database (CUDB). They obtained overall accuracy, sensitivity, and specificity of 94.0%, 99.0%, and 81.0%, respectively. In addition, the minority class in this study achieved an F1-score of 84.9% and 23.2% for AFL and VT, respectively. Yıldırım *et al.* [17] proposed a six-layer 1-D CNN for 17 classes of arrhythmia in the MITDB. This method achieved an accuracy, specificity, precision, recall, and F1-score of 91.3%, 99.4%, 89.5%, 83.9%, and 85.4%, respectively. Pławiak *et al.* [16] suggested an evolutionary neural system based on a support vector machine (SVM) for 17 classes of arrhythmia in MITDB. They used the power spectral density from raw ECG data. They obtained an overall accuracy and F1-score of 91.00% and 89.49%, respectively.

LSTM is effective in learning the temporal features of ECG signals; therefore, it has been used to classify arrhythmias. Sujadevi *et al.* [15] proposed the LSTM method to classify AFIB and NSR using the AFDB and MIT-BIH NSR databases. They demonstrated the cumbersome preprocessing of data by LSTM, such as denoising of ECG signals, and used LSTM to detect AFIB in real time. This experiment achieved an accuracy, precision, recall, and F1-score of 100.0% each for the two-class classification. Faust *et al.* [18] proposed an LSTM model for detecting and classifying AFIB and NSR, which employed a diagnostic support system for AF using only the interval between the two adjacent R peak (RR interval) features. They achieved an F1-score of 99.8% in the AFDB.

Combining CNN and LSTM is advantageous for ECG classification [53]. Chen *et al.* [19] proposed such a combination, which applied a multi-input structure to process ECG signal segments and corresponding RR intervals from MITDB. This study classified six types of ECG signals as follows: AFIB, AFL, ventricular bigeminy, pacing rhythm, and SBR. They achieved an overall accuracy of 99.3% using a fivefold cross-validation strategy. In particular, the minority class in this study exhibited an F1-score of 54.6% and 98.4% for AFL and SBR, respectively. He *et al.* [21] proposed a framework consisting of two deep neural network modules: residual convolutional and biLSTM. Their proposed model classified the nine described types of arrhythmia. The resulting F1-score of AFIB, block, premature ventricular contraction, and ST-segment were 91.4%, 87.9%, 80.1%, and 74.2%, respectively.

Most ECG rhythm classification studies have considered AFIB. However, many studies continue to classify a large number of arrhythmias. Yildirim *et al.* [22] proposed a deep neural network model that combines a CNN with LSTM to classify seven arrhythmias using the database collected by Chapman University (CU) and Shaoxing People's Hospital (SPH) [54]. This ECG database has 11 rhythm classes; however, seven rhythms were reduced in the study because of the small number of cases in some rhythm classes. This study achieved an overall accuracy of 92.2%. Although the overall accuracy was high, the sensitivity and precision of AFL were 25.0% and 32.0%, respectively. This could be attributed to the low number of data points in this class. Murat *et al.* [24] proposed a framework for both the features obtained from deep learning model layers and clinical ECG features to detect cardiac rhythms. The results of this study were used in the database collected by CU and SPH [54]. This study considered four rhythm classes by integrating a small number of cases in some of the 11 rhythm classes. The introduced framework achieved an accuracy of 90.3% using a random forest classifier. Although this study proposed a method obtained using the deep learning model layers, the AFIB class included AFIB and AFL, and there was no distinction between these two classes.

Although these models achieved high accuracies for their target classes, more than 50% of the studies were focused on AFIB [11]. In addition, the presented deep learning architecture has limitations in extracting unique characteristics of the ECG signal for some types of arrhythmias, such as AFL, AVR, and VT. Therefore, a novel architecture is required to extract the clear features of ECG signals for arrhythmia.

## III. Proposed Method

### A. Overview

The proposed framework consists of data processing, data augmentation for class imbalance, and arrhythmia classification steps. The three steps are shown in Fig. 1. The first step is to augment the data of minority classes. The second step is to extract the global feature using ResNet with an SE block. The third step uses biLSTM to extract the sequential feature from the global features. After completing these steps, we calculated the classwise probabilities using the sequential features.

### B. Preprocessing

Each record in the MITDB contains ECG lead II, lead V1, and lead V5. Lead II is typically been used in arrhythmia detection [17], [19], and we used only lead II in the ECG signal because it is commonly used in all patients. ECG data were resampled at a frequency of 256 Hz. Subsequently, a third-order Butterworth filter was applied to correct the baseline wandering. Next, because the labels of the ECG rhythm in MITDB are a set of annotations presented from a certain marker to the next marker, each type of ECG data was extracted according to the markers, each containing a 10-s recording corresponding to each label.

The employed ECG signal values were not identical because each signal had different amplitude scaling and vanishing offset effects. To eliminate these effects, we used normalization, that is, the signals are scaled to an identical level. Each ECG segment was normalized using the Z-score normalization method.

### C. Data Augmentation

To compensate for the class imbalance, data-, algorithm-, and hybrid-level approaches can be used. The data-level approach focuses on the lack of sufficient training data for classification [55]. However, the algorithm-level approach is associated with the failure of algorithms to optimize

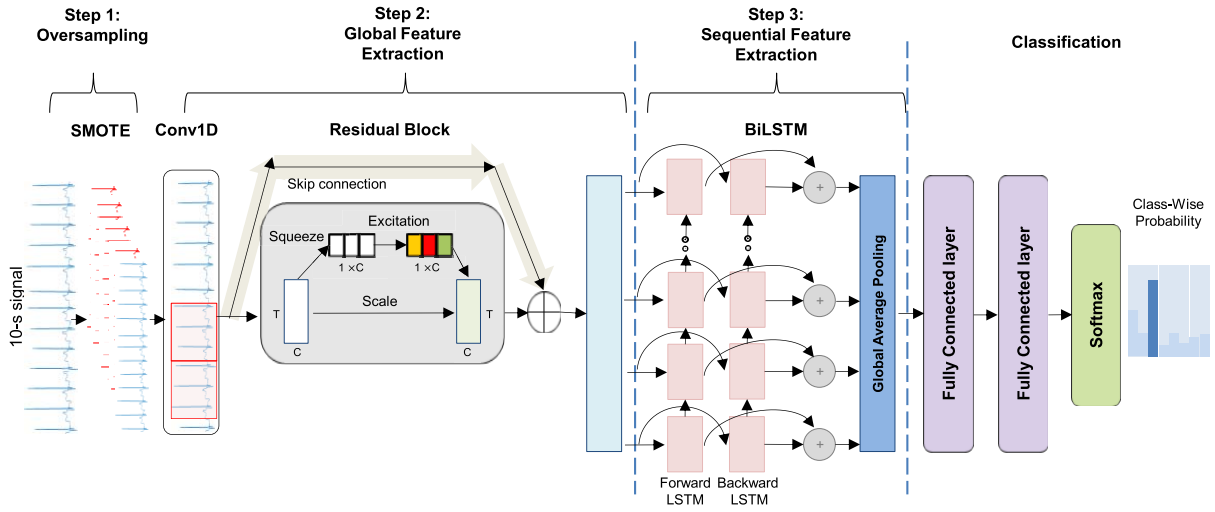


Fig. 1. Overview of the proposed arrhythmia classification. Our proposed framework consists of three steps. Step one augments data using SMOTE. The signal of SMOTE consists of a 10-s long signal (indicated by blue) and the augmented data (represented by red). The signal width represents the amplitude, and the vertical axis is time. SMOTE is applied only to the minority class. Step two is to extract global features. The convolution layer is first applied to the processed data. The red rectangular box denotes the kernel of the convolution layer in Conv1D. Subsequently, the processed data are inserted into the ResNet section to extract the global features. In the residual block, the white rectangular box within the white rectangular box indicates the feature map, which processes features recalibration. The feature map is represented as $T \times C$. $T$ refers to time points, and $C$ refers to channels. The global features extracted using ResNet are inserted into the biLSTM section for temporal feature extraction. Subsequently, the sequential feature is classified in the fully connected and Softmax layers. For example, the classification step computes the probability of each class in eight classes using MITDB.

learning for target evaluation criteria in disproportionate cases [56]. Finally, the hybrid-level approach integrates data- or algorithm-level approaches to solve the imbalance problem [56]. The employed database has a nonuniform distribution among the individual classes. That is, it includes an approximately normal class of rhythm that belongs to the majority class, and the remaining rhythms belong to other minority classes. In this case, adding multiple class instances to the dataset, in a relatively balanced proportion of the data class composition, is effective in compensating for the data imbalance [57].

We applied SMOTE, which is widely used in the data-level approach, to solve the class imbalance problem. In SMOTE, a neighborhood is first defined for each element of the minority class, identifying the $K$-nearest neighbors ($K$NNs). When a sample $x$ represents the ECG signal, $i$ is the instance of the minority class under consideration, and $j$ represents a randomly selected instance from the $K$NN of the minority class $x^i$. $\delta$ represents a vector in which every element is a random number in [0, 1]. It is used to generate a synthetic instance of two original instances $x^i$ and $x^j$, which are also known as the primary and assistant reference instances, respectively. Synthetic data were generated as follows:

$$x^{\text{synthetic}} = x^i + (x^j - x^i) \times \delta. \tag{1}$$

In this study, the nearest neighbor number of the minority class was 5 to generate synthetic samples. A large $K$ can be highly error-prone; thus, it generally chooses $K$ in the range of 4–6 [33]. Moreover, many studies typically have a $K$ set to 5 [58]. In particular, there was no statistical difference in the performance using various $K$ parameters [59]. That is, parameter $K$ did not significantly affect the performance. In the next step, $j < K$ elements of the neighborhood are randomly selected and used to construct new samples using interpolation [33].

We classified AFIB, AFL, AVR, SVT, VT, SBR, and Other classes by dividing each class with the largest number into the majority and the classes that do not have the largest number into a minority. Then, we generated multimajority that adversely affects performance. Therefore, NSR was considered the majority class [60]. When synthesizing the minority class using SMOTE, a small number of minority classes invaded other classes, increasing the possibility of generating data that confuses learning [61]. Therefore, we generated data using the following formula:

$$((N^m \div N^a) \div C) \times N^a) + N^a \tag{2}$$

where $N$ is the number of classes, $m$ is the majority class, $a$ is the class with the largest samples among minority classes, and $C$ is the number of arrhythmia classes. For example, using eight types of arrhythmias from the MITDB, $C$ was 8. In addition, $m$ is the NSR class, and $a$ is the AFIB class because the AFIB class has the largest samples compared to other minority classes, $N^m$ is the number of the NSR class, and $N^a$ is the number of the AFIB class.

The imbalance ratio of the minority classes was kept constant to maintain the data ratio of each class constant [62]. Thus, 1000 samples were prepared based on the AFIB class, which is the second largest class.

To demonstrate the advantages of SMOTE by directly comparing its performance with those of other augmentation methods, we applied ROS, GAN, and ADASYN instead of SMOTE using MITDB.

*1) ROS:* Samples from a minority class are randomly selected and added to the minority class in the training dataset. This approach was repeated until the desired class distribution

TABLE II
DETAILS OF LAYERS AND HYPERPARAMETERS IN THE PROPOSED MODEL USING MITDB

| Layers | Type | Number of filters | Kernel size | Stride | Activation function | Output shape |
|---|---|---|---|---|---|---|
| 1 | Input | - | - | - | - | 2560 × 1 |
| 2 | Conv 1D | 32 | 7 | 1 | Leaky relu | 2560 × 32 |
| 3-6 | Residual block | 32 | 7 | 1 | Leaky relu | 2560 × 32 |
| 7 | Conv 1D | 64 | 7 | 2 | Relu | 1280 × 64 |
| 8-11 | Residual block | 64 | 7 | 1 | Leaky relu | 1280 × 64 |
| 12 | Conv 1D | 96 | 7 | 2 | Relu | 640 × 96 |
| 13-16 | Residual block | 96 | 7 | 1 | Leaky relu | 640 × 96 |
| 17 | Conv 1D | 128 | 7 | 2 | Relu | 320 × 128 |
| 18-21 | Residual block | 128 | 7 | 1 | Leaky relu | 320 × 128 |
| 22 | Conv 1D | 128 | 7 | 1 | Relu | 320 × 128 |
| 23 | BiLSTM | 64 | - | - | - | 128 × 64 |
| 24 | Global average pooling | - | - | - | - | 64 |
| 25 | Fully connected layer | 333 | - | - | Relu | 333 × 37 |
| 26 | Fully connected layer | 37 | - | - | Relu | 37 × 8 |
| 27 | Classification | - | - | - | Softmax | 8 × 1 |

was achieved in the training dataset, such as an equal split across classes.

*2) Conditional GAN:* This is a GAN whose generator and discriminator are conditioned during training by using additional information [63]. The network architecture is shown in Fig. 1. The batch size and the initial learning rate were set as 128 and 0.001, respectively. In addition, we used the Adam optimizer [64].

*3) ADASYN:* We calculate ratio $a$, which is the number of samples in the minority class after resampling divided by the number of samples in the majority class. Next, we defined $K$NN as five, as used in SMOTE.

### D. ResNet With SE Block for Global Feature Extraction

Table II lists details of the kernel sizes, the number of filters, stride sizes, and output sizes for each layer. Feature extraction is a critical step in the classification of processed ECG signals. The proposed network comprises a global feature extraction and learning part, and a sequential feature extraction and learning part, as illustrated in Fig. 1.

Fig. 2 shows that, in the global feature extraction part, stacked residual convolutional modules with SE blocks are used to extract the global features and compress the long-course ECG signal into considerably shorter sequences of local feature vectors. The input of this part is the processed ECG signal, which is a 2-D matrix with determined dimensions (batch size: 2560). The batch size was set to 32. The second part (batch size: 2560) indicates the signal length used in the proposed framework. The signal length was calculated at a sampling rate of 256 Hz, multiplied by 10 s. The ResNet architecture in our proposed framework was based on the architecture presented by He *et al.* [21]. The ResNet module consists of the residual and SE block. The residual block in network consist of 1-D convolutional (Conv1D) layers, batch normalization (BN) [65] layers, leaky rectified linear units (leaky ReLUs) [66] in the activation layer, and the SE block [31].

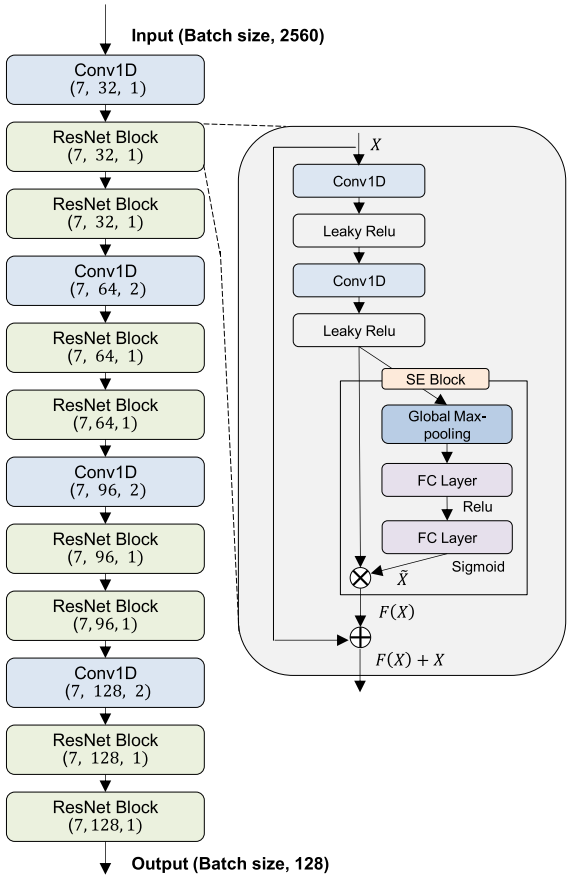


Fig. 2. Architecture of the ResNet block. 1-D convolutional layer: Conv1D. Fully connected layer: FC layer. ResNet blocks include Conv1D and SE blocks. Processed data from leaky ReLU are inserted into the SE block. In the SE block, $\tilde{X}$ indicates the output of the last FC layer. The function of $F(X)$ represents these Conv1D and SE blocks. Next, the output calculated by $F(X)$ is added $X$ via skip connection.

Mathematically, we represent these convolutional layers and SE blocks using the function $F(X)$, where $X$ is the input. The residual connection can be expressed as in (3). The residual

block performs the following computation:

$$Y = F(X) + X. \tag{3}$$

We assume that the input feature of layer $l$ is $a^{l-1} \in R \times D$, where $L$ is the number of time points in a frame and $D$ is the dimension of the features. The output $a_i^{l,c}$ of that layer is expressed as follows:

$$a_i^{l,c} = b^c + \sum_{v=1}^{D} \sum_{u=1}^{k} w_{uv}^{l,c} a_{i-\frac{k}{2}+u,v}^{l-1} \tag{4}$$

where $b^c$ is the bias term of the $c$th output feature in the set of $C$ output features ($c = 1, \ldots, C$). $k$ is the size of the kernel that slices along the time axis, and $w^{l,c}$ is the weight matrix at layer $l$ regarding the $c$th output feature. The Conv1D consists of a convolutional, BN, max-pooling [67], and leaky ReLU layer. Accordingly, there were 19 convolutional layers. As the margin of the input is lost during a convolutional operation, the input feature maps are padded before each convolutional layer such that the output has the same length as the original input. The feature maps are compressed in length only when they pass through a convolutional block. As shown in Fig. 2, each substructure consists of one SE block and a residual module. The length of the feature map is split through each of the substructures, the number of which partially depends on the input length; therefore, a longer input requires more pooling layers to compress the feature map to a certain length. Formally, a static $z \in R^C$ is generated by $U$ through its time points $T$ such that the channel of the $c$th element of $z$ is calculated by

$$z_c = F_{\text{sq}}(u_c) = \frac{1}{T} \sum_{t=1}^{T} u_c(t). \tag{5}$$

The excitation operation is intended to fully capture the channelwise dependencies. The excitation operation processes the output of the squeeze step vector $z$ to produce a vector of activations $s$, which is then used to rescale the feature maps. This activation vector $s$ is not to be confused with $s$ that was used earlier to keep track of the channels of the input $X$. Vector $s$ is calculated from the squeeze output $z$ using two fully connected layers with a bottleneck that takes the representation down to size $C/r$. The hyperparameter $r$ is referred to as the "reduction ratio." The output $s$ is presented as follows:

$$s = F_{\text{ex}}(z, W) = \sigma(W_2 \delta(W_1 z)) \tag{6}$$

where $W_1 \in \mathrm{R}^{(C/r)\times C}$ and $W_2 \in \mathrm{R}^{C\times(C/r)}$.

### *E. LSTM for Sequential Feature Extraction, Learning, and Classification*

After global feature extraction and learning, the extracted feature vectors were individually fed to a biLSTM layer (the main body of the sequential feature extraction and learning part), as shown in Fig. 3.

As the ECG signals represent the time course of the heart's electrical activity, a recurrent neural network can be typically used to process the input along the time sequence in a parameter-sharing manner, and the internal state is used to memorize the context. The unit numbers of these LSTM layers are four; that is, each local-focused global feature vector has a length of 64 units. Subsequently, all the sequence-focused global feature vectors were input into a global max-pooling layer to obtain a single global feature vector for classification.

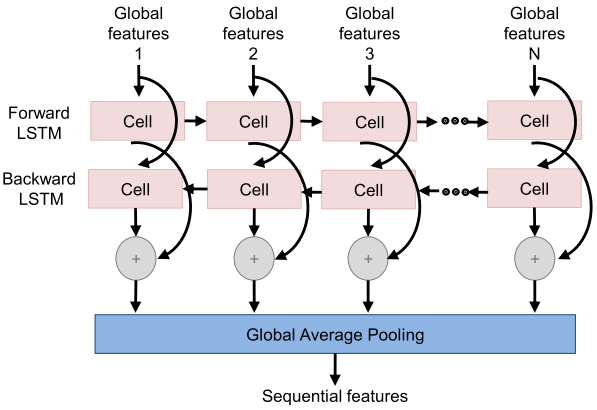


Fig. 3. Architecture of biLSTM block. LSTM: long-short term memory. biLSTM: bidirectional LSTM.

With the extracted sequence feature, the classification part learns a classifier to stratify the recordings into different classes. The classification part consists of two dense layers and two activation layers. The first dense layer contains 64 cells, whereas the second dense layer contains eight cells (corresponding to eight classes). The first activation layer after the first dense layer is an ReLU layer, which enables the classification part to accelerate the backpropagation of the gradients. The second activation layer, which is the final layer of the network, is a Softmax layer that outputs the predicted probability distribution over the eight classes. $X_t$ represents the $t_{\text{th}}$ time series value fed in the LSTM. $c_t$ represents the memory cell, which is the core of LSTM.

The equation of different cells in LSTM is as follows:

$$i_t = \sigma(W_{\text{xi}} x_t + W_{\text{hi}} h_{t-1} + b_i) \tag{7}$$
$$f_t = \sigma(W_{\text{xf}} x_t + W_{\text{hf}} h_{t-1} + b_f) \tag{8}$$
$$o_t = \sigma(W_{\text{xo}} x_t + W_{\text{ho}} h_{t-1} + b_o) \tag{9}$$
$$c_t = \tanh(W_{\text{xc}} x_t + W_{\text{hc}} h_{t-1} + b_c) \tag{10}$$
$$c_t = f_t e c_{t-1} + i_t e c_t \tag{11}$$
$$h_t = o_t e \tanh(c_t) \tag{12}$$

where $f_t$ and $o_t$ represent the $t_{\text{th}}$ input gate, the forget gate, and the output gate function, respectively. $W_{\text{xi}}$, $W_{\text{xf}}$, $W_{\text{xo}}$, and $W_{\text{xc}}$ represent the weights of the input gate, the forget gate, the input gate, and the memory cell, respectively. $W_{\text{hi}}$, $W_{\text{hf}}$, $W_{\text{ho}}$, and $W_{\text{hc}}$ represent weights from hidden layers to the input gate, the forget gate, the input gate, and the memory cell, respectively. $b_i$, $b_f$, $b_o$, and $b_c$ are the bias values of the input gate, the forget gate, the output gate, and the memory cell, respectively, where $\sigma$ is the sigmoid function. Tanh refers to the hyperbolic tangent activation function, and $J$ represents pointwise multiplication. To obtain the optimal parameters,

TABLE III
PROPOSED OPTIMAL MODEL PARAMETERS FOR ARRHYTHMIA CLASSIFICATION

| Hyperparameters | Value |
|---|---|
| Learning rate | 0.0001 |
| Optimizer | Adam [64] |
| Convolutional layer kernel size | 7 |
| No. of LSTM units | 64 |
| Batch size | 32 |

either the CNN or LSTM can use the backpropagation method to adjust the model parameters during the training process.

### F. Model Training Algorithm

As shown in Table III, the batch size was set to 32 for MITDB. The initial learning rate was set to 0.0001 using the learning rate decay strategy. The decay interval of the learning rate was 16, and it was reduced by a factor of 10 at each decay interval. This strategy increases the stability of the results and enhanced the performance of the network. We used Adam optimizer [64] for weight upgrading. Hyperparameters, including the convolutional kernel size and the number of LSTM units, were tuned in the experiments. The proposed model achieved the minimum test error among all attempted combinations when the convolutional kernel size and the number of LSTM units were 7 and 64, respectively.

## IV. EXPERIMENTS AND VALIDATION

### A. Database

In this study, we used three ECG databases; Table IV summarizes the results. The performance was evaluated by applying the proposed framework to each of the three databases, and the MITDB was used in the ablation study. In addition, the AFDB and CinC DB were tested based on an MITDB-based model to measure the generalization ability.

*1) MITDB:* This database contains 47 two-channel ambulatory ECG recordings obtained from 47 subjects collected at the MIT-BIH Arrhythmia Laboratory between 1975 and 1979. Individual recordings had a duration of approximately 30 min and a sampling rate of 360 Hz. The database includes recordings corresponding to 17 types of arrhythmic rhythms and heartbeats [68].

In this study, we selected seven classes with a high incidence of arrhythmia among 17 classes [7], the other classes were classified as "Other" classes, and a total of eight classes were classified using MITDB.

Each recording contained an annotation of the AFIB, AFL, AVR, SVT, VT, SBR, and NSR rhythms. Fig. 4 shows an example of eight different classes of arrhythmias. The details of MITDB are shown in Table I.

*2) AFDB:* AFDB contains long-term two-lead ECG records of 25 subjects with AFIB. Each data file lasted for 10 h, and the signal was sampled at a sampling rate of 250 Hz. The original ECG data were recorded at Boston's Beth Israel Hospital using ambulatory ECG recorders, with a typical recording bandwidth of approximately 0.1–40 Hz. The database includes AFIB, AFL, AVR, and NSR [68]. Details of AFDB are provided in Table II.

TABLE IV
SUMMARIZATION OF THREE DATABASE USED IN OUR EXPERIMENTS

| Dataset | No. of Subject | Sampling Rate (Hz) | ECG Class | No. of Total Samples |
|---|---|---|---|---|
| MITDB [68] | 48 | 360 | AFIB, AFL, AVR, SVT, VT, SBR, NSR, Other | 5188 |
| AFDB [68] | 25 | 250 | AFIB, AFL, AVR, NSR | 64506 |
| CinC DB [35] | 8528 | 250 | AFIB, NSR | 5128 |

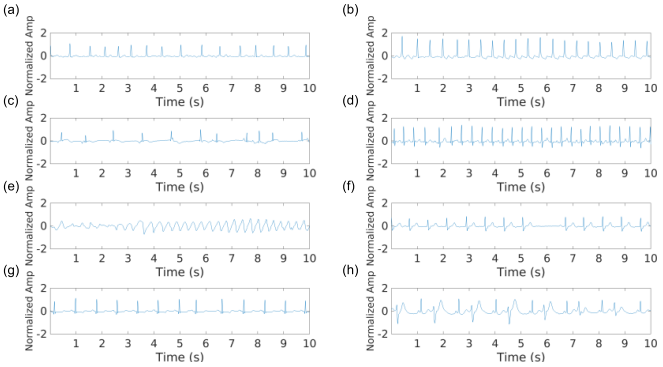


Fig. 4. Eight selected rhythms of ECG signals. Each rhythm shows a representative sample of the ECG signal from MITDB. (a) AFIB. (b) AFL. (c) AVR. (d) SVT. (e) VT. (f) SBR. (g) NSR. (h) Other.

*3) CinC DB:* The database includes AFIB, noise, NSR, and other rhythms in the short-term ECG recordings (9–61 s) with a sampling frequency of 300 Hz. We used 12 186 ECG segments, and the data consisted of 8528 subjects in the public training [35] and 3658 subjects in the private hidden test sets. Each recording was captured by an individual, who purchased one of the three generations of the AliveCor single-channel ECG device. Although there were four types of ECG data in the database, the training set in this study only used the ECG samples of NSR and AFIB because noise and other rhythms were insufficient compared to MITDB. Details of the CinC DB are provided in Table III.

### B. Experimental Setup

We used $K$-fold cross-validation to overcome the overfitting problem. In this study, we set $K = 5$ as used in many previous studies [19], [21]. The processed ECG segments were randomly shuffled using a fivefold validation process. In addition, to check the influence of $K$, we computed a tenfold cross-validation as $K = 10$ [69] and compared the performance metrics.

The original training set from MITDB indicates that 80% of the MITDB training did not apply SMOTE. The oversampled training set included data synthesized in 80% for the MITDB training. The number of data samples was set to 1000 to maintain a constant distribution of the minority classes. In addition,

the original and oversampled training sets from different ECG databases used the same procedure as MITDB.

### C. Few-Shot Learning Using Independent Database

In this experiment, models trained using the source database (MITDB) were tested on the other target databases (AFDB and CinC DB). The experiment consisted of four steps to measure the generalization ability. First, MITDB was used as the source database [68], where AFDB was the target database, and only four classes, AFIB, AFL, AVR, and NSR, were trained. As the target database, AFDB consists of four classes: Only four of the classes were selected in the source database for the response. Moreover, only AFIB and NSR classes were trained when CinC DB was used as the target database because it includes only two classes as in the case of AFDB. Second, the proposed method was used to train the source database. Third, we randomly selected $k$ samples from four categories of the target database as the AFDB. For the CinC DB, we randomly selected $k$ samples from two categories of the target database. The $k$ samples randomly selected from each target database were applied equally to the proposed method, ShallowConvNet, and DeepConvNet. Next, $k$ annotated samples are selected to constitute the support set for $k$-shot learning. Because $k$ is typically used in the range of 0–20 in the dataset [70], [71], we chose 0, 5, 10, and 20 for samples annotated with $k$. Finally, we have trained $zero$-shot, five-shot, ten-shot, and 20-shot in trained classifiers [40], [72]. The support set was used to optimize the classifier for 60 iterations, where the batch size was 1.

To compare generalization ability, we followed the same steps and applied different methods, including ShallowConvNet and DeepConvNet.

*1) ShallowConvNet [73]:* This method designed temporal convolution, spatial filter, squaring nonlinearity, a mean pooling layer, and a logarithmic activation function. This architecture is widely used in the field of biosignals because it can extract temporal and spatial features to match the characteristics of biosignals [42].

*2) DeepConvNet [43]:* This method consists of four convolution-max-pooling blocks. The first block is designed to extract the temporal and electrode characteristics of a biosignal. Therefore, DeepConvNet is widely used in the field of biosignals [43].

### D. Cross-Subject Experiment Using Same Database

In this experiment, the model trained two classes including AFIB and NSR in the AFDB of the remaining subjects that did not choose subjects evaluated one subject that chosen subject. Details of the classes for each subject in MITDB and AFDB are provided in Tables IV and V. To compare generalization ability, we followed the same steps and applied different classic machine learning methods, including SVM and linear discriminant analysis (LDA), $K$NN, multilayer perceptron (MLP), and Gaussian naive Bayes (GNB). These methods are classically used in arrhythmia classification problem [74]–[76], which does not use deep learning. Specifically, the SVM classifier used the radial basis function. The parameters $Gamma$ and $C$ were set to 0.7 and 1, respectively. The $K$NN classifier has a $K$ set to 3. MLP used the Adam optimizer [64], and the initial learning rate was set to 0.001.

### E. Evaluation Metrics

The overall precision, recall, specificity, F1-score, and G-mean were used as the performance criteria. These performance metrics are indicators of commonly used model performance [77], [78]. In particular, G-mean is a metric that measures the balance between classification performance in both majority and minority classes [79]. As in other ECG studies, the classes were imbalanced, and minority classes were particularly inadequately represented [19]. Therefore, we measured the classwise precision, recall, specificity, F1-score, and G-mean as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{13}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{14}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{15}$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{16}$$

$$\text{G-mean} = \sqrt{\text{Specificity} \times \text{Recall}} \tag{17}$$

where TP, FP, TN, and FN represent true positives, false positives, true negatives, and false negatives, respectively.

### F. Statistical Analysis

The difference between the nonoversampled and oversampled datasets using SMOTE was evaluated using the Mann–Whitney test. The difference between $K = 5$ and $K = 10$ of the cross-validation performance was also evaluated using the Mann–Whitney test. The performance metrics between the frameworks with and without SE blocks were tested using the Mann–Whitney test to evaluate the SE block effect. Moreover, the differences among oversampling methods, including SMOTE, ROS, GAN, and ADASYN, were assessed using the Kruskal–Wallis tests, with pairwise Wilcoxon tests with the least significant difference (LSD) for post hoc analyses. Next, we assessed the differences in the cross-subject performance among SVM, LDA, $K$NN, MLP, GNB, and the proposed method using the Kruskal–Wallis tests and pairwise Wilcoxon tests with LSD for post hoc analyses. A significance level of 5% ($p < 0.05$) was considered significant for all analyses.

### G. Implementation

The proposed method and baseline models were implemented using MATLAB R2020b and Python 3.7. MATLAB was used to load and preprocess the raw ECG signals, and Python was employed to implement the model using Keras v1.3.1. For training and testing, a computer with an Intel Xeon-Gold 6246 with 3.30-GHz CPU, 128-GB RAM, and an NVIDIA RTX 8000 GPU was used.

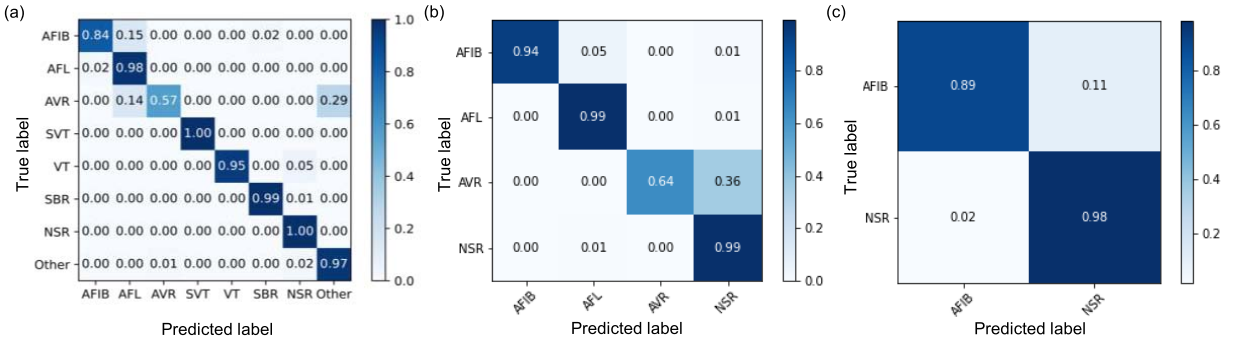


Fig. 5. Average confusion matrix of the ECG segments classified by the proposed framework. (a) MITDB. (b) AFDB. (c) CinC DB. The color bar indicates the proportion of samples per class assigned to the correct label.

TABLE V
COMPARISON OF CROSS-VALIDATION PERFORMANCE USING PROPOSED FRAMEWORK IN MITDB

| Approach | Metrics | Performance |
|---|---|---|
| 5-fold cross-validation | Precision (%) | 92.23 ± 2.95 |
| | Recall (%) | 91.23 ± 6.17 |
| | Specificity (%) | 99.82 ± 0.04 |
| | F1-score (%) | 91.69 ± 1.89 |
| | G-Mean (%) | 95.43 ± 4.56 |
| 10-fold cross-validation | Precision (%) | 86.25 ± 8.81 |
| | Recall (%) | 85.45 ± 8.29 |
| | Specificity (%) | 99.47 ± 0.27 |
| | F1-score (%) | 85.67 ± 8.60 |
| | G-Mean (%) | 88.05 ± 8.11 |

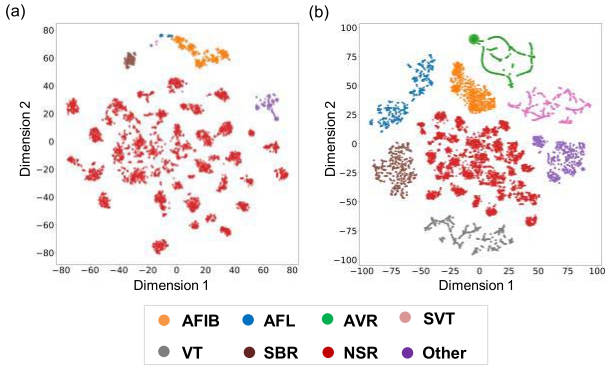


Fig. 6. Visualization via t-SNE of features extracted from the global average pooling layer of the proposed method trained using the original ECG data and ECG data augmented with SMOTE in the MITDB. (a) Features using original ECG data. (b) Features using augmented ECG data with SMOTE.

## V. RESULTS AND DISCUSSION

### A. Classification Performance of Proposed Model

The performance of the proposed framework was evaluated using MITDB, AFDB, and CinC DB. We used fivefold cross-validation to overcome the overfitting problem. Fig. 5 shows the confusion matrix for the three databases. The eight-class classification performance using MITDB achieved an overall accuracy and F1-score of 99.20% and 91.69%, respectively. In addition, four-class arrhythmia using AFDB in the proposed method achieved an overall accuracy and F1-score of 99.35% and 92.86%, respectively. Finally, the two-class arrhythmia using CinC DB in the proposed method obtained an overall accuracy and F1-score of 97.05% and 91.44%, respectively.

In addition, we compared the classification performance between the fivefold cross-validation and tenfold cross-validation approaches using MITDB to check the influence of performance according to the $K$ value.

Table V shows the performance metrics using fivefold cross-validation and tenfold cross-validation approaches. Therefore, no statistically significant difference exists among the performance metrics. The statistical result was shown in Table VI.

Next, to visualize the distribution between each class of the MITDB nonoversampled dataset and the oversampled dataset, we applied the Barnes–Hut variant of t-distributed stochastic neighbor embedding (t-SNE) [80]. Fig. 6 shows the distribution between each class of features extracted through the global average pooling layer of the proposed method using t-SNE. As a result, the AFIB and AFL classes overlapped in the nonoversampled dataset. However, despite the increase in the number of minority classes when using the SMOTE method, these classes were well distributed. Examples of the ECG signals in the original training set and the oversampled training set using SMOTE are shown in Fig. 2.

We compared the performances of various oversampling methods using the fivefold cross-validation in the proposed framework based on MITDB to investigate the effect of SMOTE. We conducted the Kruskal–Wallis test with LSD post hoc tests. Fig. 7 shows a statistical comparison between the oversampling methods, including SMOTE, ROS, GAN, and ADASYN. The results indicate that the performance of the proposed model using SMOTE was statistically higher in terms of recall, specificity, F1-score, and G-mean ($p < 0.05$). Furthermore, post hoc analysis showed that the performance of SMOTE was statistically higher than that of ROS, GAN, and ADASYN in terms of recall. The SMOTE of specificity statistically differed from GAN and ADASYN. In addition, the F1-score in SMOTE was statistically higher than that of ROS and GAN. In the G-mean, SMOTE was statistically higher than that of GAN and ADASYN.

TABLE VI
COMPARISON BETWEEN THE OVERALL PERFORMANCE METRICS OF EACH MODEL IN ORIGINAL AND OVERSAMPLED TRAINING SETS USING THE MITDB

| SMOTE | ResNet | BiLSTM | SE block | Precision (%) | Recall (%) | Specificity (%) | F1-score (%) | G-Mean (%) |
|---|---|---|---|---|---|---|---|---|
| X | O | X | X | 79.00 ± 4.08 | 76.38 ± 2.75 | 99.51 ± 0.17 | 76.91 ± 2.73 | 80.72 ± 1.79 |
| X | O | X | O | 86.08 ± 2.23 | 82.33 ± 5.20 | 99.57 ± 0.33 | 83.60 ± 3.17 | 90.54 ± 5.69 |
| X | X | O | X | 10.45 ± 0.0 | 12.50 ± 0.04 | 87.50 ± 0.01 | 11.39 ± 0.01 | 33.07 ± 0.01 |
| X | O | O | X | 79.89 ± 3.93 | 75.86 ± 2.61 | 99.49 ± 0.18 | 77.22 ± 2.98 | 80.45 ± 1.56 |
| X | O | O | O | 88.63 ± 3.68 | 84.94 ± 7.12 | 99.64 ± 0.10 | 86.58 ± 3.04 | 92.00 ± 5.01 |
| O | O | X | X | 86.05 ± 3.51 | 84.77 ± 6.05 | 99.65 ± 0.11 | 84.21 ± 3.94 | 88.76 ± 5.35 |
| O | O | X | O | 88.68 ± 3.13 | 86.98 ± 3.31 | 99.68 ± 0.08 | 87.57 ± 0.70 | 93.11 ± 4.61 |
| O | X | O | X | 22.67 ± 0.04 | 29.45 ± 7.34 | 89.43 ± 8.50 | 17.50 ± 8.13 | 51.32 ± 3.12 |
| O | O | O | X | 88.60 ± 4.65 | 82.80 ± 3.33 | 99.59 ± 0.07 | 84.36 ± 3.06 | 87.82 ± 4.06 |
| O | O | O | O | 92.23 ± 2.95 | 91.23 ± 6.17 | 99.82 ± 0.04 | 91.69 ± 1.89 | 95.43 ± 4.56 |

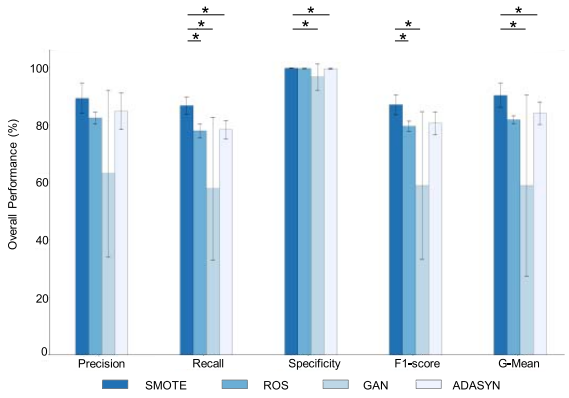


Fig. 7. Statistical comparison between overall performance metrics in oversampling methods, including SMOTE, ROS, GAN, and ADASYN. The result is presented as the overall mean ± standard deviation. $*p < 0.05$ (the Kruskal–Wallis test with LSD post hoc).

The reasons for the higher performance of SMOTE are given as follows. First, the ROS result is low because it may not have the effect of significantly increasing awareness of the minority class [46]. Second, although ADASYN is an extended version of SMOTE, the performance of ADASYN is poorer than SMOTE. SMOTE may outperform ADASYN when the degree of class imbalance increases [52]. Finally, while GAN produces data that appear comparable to the original data, these networks are difficult to train and often considerably complex to produce results [50]. Therefore, the standard deviation values were highly variable. Despite the use of various data augmentation methods, SMOTE is more suitable for classifying arrhythmia with a large number of classes and high class imbalance. Consequently, SMOTE was the best oversampling method for the proposed framework. A statistical comparison of the oversampling methods is shown in Table VII.

### *B. Ablation Study*

The proposed framework consists of three components: 1) SMOTE; 2) ResNet combined with SE block architecture; and 3) the biLSTM architecture. Moreover, we discuss ablation experiments using MITDB to demonstrate the role and effectiveness of components. Table VI lists the overall performance of the proposed components.

*1) SMOTE:* This was used to solve the class imbalance problem. Table VI shows that the oversampled training set using SMOTE achieved an overall precision of 92.28%, whereas that of the original training set was 88.63%. The original and oversampled training sets achieved an overall recall of 84.93 and 91.23%, respectively. The overall specificity of the original training set was 99.64%, whereas that of the oversampled training set achieved 99.83%. The F1-score of the oversampled training set obtained 91.69%, whereas that of the original training set was 86.59%. The G-mean scores of the oversampled and original training sets were 95.43% and 92.00%, respectively. In summary, all performance metrics, except for the specificity, significantly increased in the oversampled training set using SMOTE. The statistical results are presented in Table VIII.

Table VII shows that the classification performance of the proposed framework using SMOTE was higher than that of the original training set in terms of precision, recall, F1-score, and G-mean for AFIB, AFL, AVR, SVT, and SBR. In particular, in the proposed framework using SMOTE, the F1-score of AVR was 61.54%, which is higher than the obtained F1-score of 40.00% achieved by the proposed framework using the original training set. In addition, the G-mean was higher for the proposed framework using the oversampled training set than for the proposed framework using the original training set in all classes. When the previous probabilities of the classes differ considerably, such metric comparisons may be misleading. For example, it is straightforward to create a classifier with an accuracy of 99% when the dataset has a majority class with 99% of the total number of cases by simply labeling every new case as the majority class. In this case, misleading can be prevented by matching the data of the minority class with the proportion of the majority class [81]. However, although the classification performance of AVR increased according to all the metrics, misclassification between AFL and AVR and between AVR and Other increased. The results indicate that, when SMOTE is used to augment AVR data, the number of original samples of the AVR class is small, and this small number of samples is located at the boundary between the different classes. Therefore, the misclassification would have increased because the newly synthesized samples were located near the boundary between the Other and AFL classes [82].

The SMOTE approach generates more samples in minority classes to achieve class balance. The proposed method trained

TABLE VII
COMPARISON BETWEEN THE CLASSWISE CLASSIFICATION PERFORMANCES USING PROPOSED FRAMEWORK IN MITDB

| Oversampling methods | Class | Precision (%) | Recall (%) | Specificity (%) | F1-score (%) | G-Mean (%) |
|---|---|---|---|---|---|---|
| Non-oversampled | AFIB | 81.25 ± 9.83 | 69.64 ± 1.09 | 99.86 ± 0.09 | 75.00 ± 5.18 | 83.39 ± 0.68 |
| | AFL | 97.26 ± 0.57 | 97.62 ± 1.78 | 99.75 ± 0.05 | 97.44 ± 1.00 | 98.68 ± 0.91 |
| | AVR | 37.50 ± 22.61 | 42.86 ± 35.42 | 99.92 ± 0.05 | 40.00 ± 20.50 | 65.44 ± 30.82 |
| | SVT | 100.00 ± 0.00 | 81.82 ± 24.50 | 100.00 ± 0.00 | 90.00 ± 16.33 | 90.45 ± 14.35 |
| | VT | 100.00 ± 0.00 | 95.00 ± 10.00 | 100.00 ± 0.00 | 97.44 ± 5.71 | 97.47 ± 5.36 |
| | SBR | 98.24 ± 1.14 | 99.40 ± 1.21 | 99.95 ± 0.03 | 98.82 ± 0.73 | 99.67 ± 0.61 |
| | NSR | 99.58 ± 0.22 | 99.78 ± 0.11 | 97.84 ± 1.12 | 99.68 ± 0.07 | 98.81 ± 0.52 |
| | Other | 95.22 ± 2.49 | 93.36 ± 2.92 | 99.81 ± 0.11 | 94.28 ± 0.95 | 96.53 ± 1.48 |
| SMOTE | AFIB | 82.14 ± 9.96 | 83.64 ± 7.48 | 99.84 ± 0.11 | 82.88 ± 7.29 | 91.31 ± 4.13 |
| | AFL | 98.34 ± 0.68 | 98.16 ± 1.29 | 99.85 ± 0.06 | 98.25 ± 0.74 | 99.00 ± 0.65 |
| | AVR | 66.67 ± 27.42 | 57.14 ± 26.42 | 99.97 ± 0.04 | 61.54 ± 25.67 | 75.58 ± 30.55 |
| | SVT | 100.00 ± 0.00 | 100.00 ± 0.00 | 100.00 ± 0.00 | 100.00 ± 0.00 | 100.00 ± 0.00 |
| | VT | 100.00 ± 0.00 | 95.00 ± 9.25 | 100.00 ± 0.00 | 97.44 ± 5.71 | 97.47 ± 5.36 |
| | SBR | 99.40 ± 0.02 | 99.40 ± 1.21 | 99.98 ± 0.28 | 99.40 ± 0.62 | 99.69 ± 0.61 |
| | NSR | 99.85 ± 0.09 | 99.63 ± 0.17 | 99.25 ± 0.48 | 99.74 ± 0.08 | 99.44 ± 0.22 |
| | Other | 91.85 ± 3.30 | 96.88 ± 1.97 | 99.65 ± 0.16 | 94.30 ± 1.46 | 98.26 ± 0.95 |

unique feature representations from a balanced database. In addition, this approach causes the classifier to build larger decision regions that contain nearby minority class points [33]. Therefore, the proposed framework reduces the failure of predicting the correct class for the minority class. Therefore, the proposed method using SMOTE achieved statistically higher performance than that of the original training dataset and other oversampling methods in the proposed method. Moreover, SMOTE directly contributes to improving the arrhythmia classification performance of the proposed model by increasing the amount of trained data.

*2) ResNet:* Table VI shows that, in the absence of biLSTM and the SE block, the ResNet framework obtained a precision, recall, specificity, F1-score, and G-mean of 79.00 ± 4.08%, 76.38 ± 2.75%, 99.51 ± 0.17%, 76.91 ± 2.73%, and 80.72 ± 1.79%, respectively. In absence of ResNet and SE block, biLSTM showed a low classification performance of 10.45 ± 0.00, 12.50 ± 0.04, 87.50 ± 0.01, 11.39 ± 0.01, and 33.07 ± 0.01 of the precision, the recall, the specificity, the F1-score, and the G-mean in the original training set. ResNet combined with biLSTM slightly increased each performance metric over the ResNet framework. Thus, ResNet satisfactorily extracts the global features of arrhythmia in all patients [31].

*3) BiLSTM:* We removed the biLSTM framework to examine its effect. The biLSTM framework achieved lower performance than the ResNet framework. We added SMOTE to the biLSTM framework, which increased all the performance metrics; however, the classification performance remained low. Results indicated that biLSTM achieved a low performance when its input was in an extremely long range [83]. Faust *et al.* [18] used the input of the RR interval to extract features for temporal changes. Khan and Kim [77] used the input of dimension-reduction data using principal component analysis from the ECG signal. The results of these studies were incorporated into the features, whereas our method used long-range ECG signals. Faust *et al.* [18] and Sujadevi *et al.* [15] used the binary classification to classify only AFIB and NSR; thus, the chance level was higher than that of the proposed method. Moreover, although Sujadevi *et al.* [15] used processed ECG signals, only 25 samples were used for binary classification. By contrast, our biLSTM model has a long-range dependency problem because it uses a greater number of classes. Although ResNet combined with biLSTM exhibited a similar performance when we compared the ResNet framework, biLSTM combined ResNet and the SE block framework to increase precision, recall, F1-score, and G-mean compared with those of the ResNet combined with the SE block. Therefore, biLSTM plays an important role in extracting associations between individual features by summarizing global features into sequential features for ECG classification [21], [84].

*4) SE Block:* When we added the SE block to the ResNet framework, the precision, recall, F1-score, and G-mean increased above 6% compared to those of the ResNet framework alone. In addition, when we added the SE block and biLSTM, all overall metrics increased by approximately 2% in the ResNet combined with the SE block and biLSTM framework compared with those of the ResNet combined with the biLSTM framework. This was equally high for both ResNet combined a biLSTM framework with and without SMOTE. In addition, we compared the results with and without the SE block to evaluate the effect of the SE block. Thus, a statistically significant difference exists in the performance between with and without SE blocks. The results are presented in Table IX. In addition, the role of the SE block is to recalibrate channel-specific feature responses by explicitly modeling the interdependencies between channels in ResNet [31]. Moreover, the SE block further strengthens the representation of the features in ResNet.

### C. Few-Shot Learning Using Independent Database

We tested two independent databases after training the proposed model using the source database [19] to measure the generalization ability. Table VIII shows generalization performance for the three deep learning models under two different independent database conditions.

TABLE VIII
COMPARISON OF FEW-SHOT LEARNING USING AN INDEPENDENT DATABASE

| Test Database | K-shot | Method | Precision (%) | Recall (%) | Specificity (%) | F1-score (%) | G-Mean (%) |
|---|---|---|---|---|---|---|---|
| AFDB | Zero-shot | ShallowConvNet [73] | 22.15 | 15.11 | 74.19 | 16.54 | 11.23 |
| | | DeepConvNet [43] | 22.15 | 19.82 | 74.25 | 16.11 | 12.18 |
| | | Proposed method | 30.76 | 25.61 | 77.59 | 22.58 | 48.38 |
| | 5-shot | ShallowConvNet [73] | 23.73 | 14.73 | 73.35 | 17.91 | 11.49 |
| | | DeepConvNet [43] | 28.67 | 31.50 | 80.73 | 22.69 | 31.71 |
| | | Proposed method | 40.80 | 37.14 | 82.39 | 24.04 | 52.79 |
| | 10-shot | ShallowConvNet [73] | 23.04 | 14.73 | 73.17 | 17.97 | 11.06 |
| | | DeepConvNet [43] | 26.89 | 32.40 | 78.91 | 23.56 | 30.15 |
| | | Proposed method | 34.17 | 39.51 | 80.45 | 28.93 | 54.93 |
| | 20-shot | ShallowConvNet [73] | 29.63 | 15.39 | 75.23 | 8.39 | 14.76 |
| | | DeepConvNet [43] | 30.24 | 30.71 | 80.05 | 28.48 | 33.04 |
| | | Proposed method | 38.28 | 58.67 | 83.50 | 30.30 | 67.77 |
| CinC DB | Zero-shot | ShallowConvNet [73] | 50.02 | 56.21 | 56.21 | 63.28 | 35.80 |
| | | DeepConvNet [43] | 50.22 | 54.20 | 54.20 | 11.24 | 34.21 |
| | | Proposed method | 74.82 | 60.91 | 60.91 | 63.99 | 48.61 |
| | 5-shot | ShallowConvNet [73] | 50.10 | 56.44 | 56.44 | 11.60 | 35.89 |
| | | DeepConvNet [43] | 50.58 | 54.22 | 54.22 | 13.18 | 35.22 |
| | | Proposed method | 61.83 | 66.87 | 66.87 | 63.32 | 64.54 |
| | 10-shot | ShallowConvNet [73] | 50.10 | 56.44 | 56.44 | 11.62 | 35.90 |
| | | DeepConvNet [43] | 50.25 | 54.90 | 54.90 | 12.07 | 35.38 |
| | | Proposed method | 59.61 | 70.39 | 70.39 | 57.45 | 70.27 |
| | 20-shot | ShallowConvNet [73] | 50.11 | 56.44 | 56.44 | 11.64 | 35.90 |
| | | DeepConvNet [43] | 49.94 | 31.42 | 31.42 | 11.40 | 25.35 |
| | | Proposed method | 73.01 | 78.30 | 78.30 | 75.19 | 77.31 |

First, we tested the AFDB using the source database. The $k$-shot classification performances were compared to measure the generalization ability of the AFDB. The precision, the recall, the F1-score, and the G-mean in the 20-shot of the proposed method were 38.28%, 58.67%, 83.50%, 30.30%, and 67.77%, respectively, while the precision, the recall, the specificity, the F1-score, and the G-mean in the 20-shot of the ShallowConvNet were 29.63%, 15.39%, 75.23%, 8.39%, and 14.76%, respectively. The precision, the recall, the specificity, the F1-score, and the G-mean achieved by DeepConvNet were 30.24%, 30.71%, 80.05%, 28.48%, and 33.04%, respectively. In addition, DeepConvNet achieved higher performance than ShallowConvNet, whereas its performance was lower than that of the proposed method. In addition, the overall performance metrics of $zero$-shot were the highest for the proposed model trained on the source dataset. In addition, all the overall performance metrics were the highest for the proposed method when we added a different number of support datasets. Unlike the proposed method, the performances of ShallowConvNet and DeepConvNet did not increase, even when the number of support datasets was increased.

Moreover, we tested CinC DB from the source database to classify AFIB and NSR. The precision, recall, specificity, F1-score, and G-mean in the 20-shot of the proposed method from CinC DB were 73.01%, 78.30%, 78.30%, 75.19%, and 77.31%, respectively. In addition to the AFDB, our proposed method obtained the highest performance in all the overall performance metrics when we added a different number of support datasets. The F1-score in DeepConvNet is relatively low compared to the precision and recall criteria. DeepConvNet classified most of them as NSR and could not properly differentiate between AFIB and NSR. Therefore, the precision and recall scores were skewed to one side, and F1-score, as the harmonic mean, was relatively low.

TABLE IX
STATISTICAL COMPARISON USING AFDB

| Method | F1-score (%) | $p$-value |
|---|---|---|
| SVM | 25.62±21.81 | <0.001 |
| LDA | 41.78±5.90 | <0.001 |
| $K$NN | 48.15±11.83 | 0.003 |
| MLP | 51.53±13.65 | 0.020 |
| GNB | 21.77±11.98 | <0.001 |
| Proposed method | 79.91±24.42 | - |

### *D. Cross-Subject Experiment Using Same Database*

We compared the two-class classification performance using the proposed method and typical machine learning methods in the AFDB to investigate the generalization ability. Table IX shows a statistical comparison of the overall F1-score between the proposed method and typical machine learning methods, including SVM, LDA, $K$NN, MLP, and GNB. The result indicates that the performance of the proposed method was statistically higher in terms of F1-score ($p < 0.001$). Furthermore, the post hoc analysis showed that the performance of the proposed method was statistically higher than that of SVM, LDA, $K$NN, MLP, and GNB in terms of F1-score. The statistical results are presented in Fig. 3.

### *E. Comparison With State-of-the-Art Methods*

Table X presents a comprehensive performance comparison between the proposed method and existing methods for

TABLE X
COMPARISON OF CLASSIFICATION PERFORMANCE OF THE PROPOSED MODEL AND STATE-OF-THE-ART METHODS

| Authors | Year | Feature | Classifier | Augmentation | Database | No. of Class | Performance |
|---|---|---|---|---|---|---|---|
| **Andreotti et al. [14]** | 2017 | Raw data | CNN | - | CinC DB | 4 | F1-score: 83.00 |
| **Acharya et al. [9]** | 2017 | Raw data | Eleven-layer Deep CNN | - | MITDB + AFDB + CUDB | 4 | Acc: 94.50<br>F1-score: 71.50 |
| **Pławiak et al. [16]** | 2018 | PSD | Evolutionary neural system (based on SVM) | - | MITDB | 17 | Acc: 91.00<br>F1-score: 89.49 |
| **Yıldırım et al. [17]** | 2018 | Raw data | Sixteen-layer deep CNN | - | MITDB | 17 | Acc: 91.33<br>F1-score: 85.19 |
| **Chen et al. [19]** | 2020 | Rawdata + RRI | CNN + LSTM | - | MITDB | 6 | Acc: 99.25<br>F1-score: 90.82 |
| **He et al. [21]** | 2020 | Raw data | ResNet + biLSTM | ROS | CPSC | 9 | F1-socre: 80.60 |
| **Yıldırım et al. [22]** | 2020 | Raw data | CNN + LSTM | - | CU, SPH | 7 | Acc: 92.24<br>F1-score: 80.04 |
| **Jin et al. [85]** | 2020 | Multi-domain features | TAC-LSTM | - | AFDB | 2 | Acc: 98.51<br>F1-score: 98.15 |
| **Petmezas et al. [86]** | 2021 | Raw data | CNN + LSTM with FL | - | AFDB | 4 | F1-score: 80.89 |
| **Proposed method** | - | Raw data | ResNet with SE block + biLSTM | SMOTE | MITDB | 8 | Acc: 99.20<br>F1-score: 91.69 |
| | | | | | AFDB | 4 | Acc: 99.35<br>F1-score: 92.86 |
| | | | | | CinC DB | 2 | Acc: 97.05<br>F1-score: 93.47 |

MIT-BIH atrial fibrillation database, AFDB. Creighton university ventricular tachyarrhythmia database, CUDB. MIT-BIH arrhythmia database, MITDB. China physiological signal challenge, CPSC. Champman university, CU. Shaoxing people's hospital, SPH. Power spectral density, PSD. RR interval, RRI. Principal component analysis, PCA. Random oversampling, ROS. Generative oversampling method, GenOME. Synthetic minority oversampling technique, SMOTE. Residual network, ResNet. Bidirectional long short-term memory biLSTM. ResNet combined with biLSTM, ResNet + biLSTM. Twin-attentional convolutional LSTM, TAC-LSTM. Focal loss, FL. Accuracy, Acc.

arrhythmia classification. The word "Augmentation" in the table header indicates that information on the augmentation method is provided. In addition, we describe the corresponding method and report the performance of each method using an overall accuracy and F1-score.

Andreotti *et al.* [14] used the CinC DB to classify four rhythms, and they achieved 83.00% F1-score. Acharya *et al.* [87] used 11-layer deep CNN classifier on a combination of MITDB, AFDB, and CUDB. Furthermore, accuracy and F1-score of 94.50% and 71.50% were achieved. Pławiak *et al.* [16] constructed an evolutionary neural system to detect 17 types of arrhythmias. Their model yielded an accuracy and F1-score of 91.00% and 89.49%, respectively. Yıldırım *et al.* [17] proposed a 16-layer deep CNN to classify 17 types of arrhythmias. They obtained an F1-score of 85.19%. Chen *et al.* [19] proposed a deep learning model combining a CNN with LSTM to classify six types of arrhythmias. Their model used raw data and RR intervals and achieved an F1-score of 90.82%. He *et al.* [21] combined ResNet with biLSTM to detect nine types of arrhythmias. They obtained an F1-score of 80.60%. Yildirim *et al.* [22] used the database collected by the CU and SPH to classify seven types of arrhythmias and achieved an F1-score of 80.04%. Jin *et al.* [85] used AFDB to classify two classes. They proposed a twin-attentional convolutional LSTM using multidomain features. Their method obtained an F1-score of 98.15%. Petmezas *et al.* [86] used AFDB to detect foul arrhythmic rhythms and achieved an F1-score of 80.89%.

MITDB, AFDB, and CinC DB were used to test the performance of the proposed framework. Our proposed framework in MITDB, AFDB, and CinC DB obtained an F1-score of 91.69%, 92.86%, and 93.47%, respectively. Although it is difficult to directly compare in MITDB, our proposed method outperformed existing studies because no significant difference existed in the chance level [88]–[90]. We trained and tested different accessible ECG databases for a direct comparison with the proposed model and another model. When using AFDB, the obtained F1-score of our method is higher than that obtained by Petmezas *et al.* [86]. The F1-score of our method is higher than that obtained by Andreotti *et al.* [14], who used CinC DB. ResNet was used with the SE block to extract the global features for rescaling the ECG data, as the SE block is essential in extracting the global features [31] and provides higher performance.

Table XI shows the F1-score of specific classes obtained by the proposed and four comparison methods. The proposed method achieved a higher F1-score in certain minority classes, including AFL and SBR, compared with the results presented by Chen *et al.* [19]. The F1-score for AFL, VT, SBR, and NSR were higher than those reported by Acharya *et al.* [9]. The proposed method achieved a higher F1-score for AFIB, AFL, SVT, and SBR than that reported by Yildirim *et al.* [22]. In addition, the proposed method using AFDB achieved an F1-score of 95.78%, 99.22%, 78.26%, and 99.50% for AFIB, AFL, AVR, and NSR. Therefore, the F1-score of AFL, AVR, and NSR obtained using the proposed method were higher than those reported by Petmezas *et al.* [86]. The F1-score of the proposed method in CinC DB was 88.63% and 98.30%, respectively.

Thus, the proposed method satisfactorily extracts features from the eight classes and learns. In real-world clinical environments, sufficient datasets are often not available; thus, high performance is highly important. Moreover, our model outperforms the state-of-the-art models in arrhythmia classification, which may make it more appropriate for real-world scenarios than existing models.

TABLE XI
COMPARISON OF CLASSWISE F1-SCORE BETWEEN THE PROPOSED MODEL USING THREE DATABASE AND STATE-OF-THE-ART METHODS

| Authors | Year | Database | No. of Class | F1-score | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | AFIB | AFL | AVR | SVT | VT | SBR | NSR |
| **Acharya et al. [9]** | 2017 | MITDB + AFDB + CUDB | 4 | 96.48 | 84.88 | - | - | 23.15 | - | 80.88 |
| **Andreotti et al. [14]** | 2017 | CUDB | 4 | 78.00 | - | - | - | - | - | 93.00 |
| **Chen et al. [19]** | 2020 | MITDB | 6 | **97.51** | 54.55 | - | - | - | 98.35 | 99.64 |
| **Yıldırım et al. [22]** | 2020 | CU + SPH | 7 | 93.45 | 29.17 | - | 83.72 | - | 98.73 | 99.64 |
| **Petmezas et al. [86]** | 2021 | AFDB | 4 | 97.05 | 88.90 | 42.18 | - | - | - | 98.42 |
| **Proposed method** | - | MITDB | 8 | 82.88 | **98.25** | **61.54** | **100.00** | **99.44** | **99.40** | **99.74** |
| | | AFDB | 4 | 95.78 | **99.22** | **78.26** | - | - | - | **99.50** |
| | | CinC DB | 2 | 88.63 | - | - | - | - | - | 98.30 |

### *F. Limitation*

The proposed model exhibited the highest classification performance among all comparison models for some types of arrhythmias. However, the results were obtained using an independent database, and in the case of using various few-shot learnings to evaluate generalization ability, desirable results cannot be guaranteed. Its performance can falter when it encounters new data in the field of medicine. Therefore, an advanced and more generalized model is required.

SMOTE has the disadvantage that the augmentation effect can be minimized because baseline wander and noise could be added to the rhythm data. However, SMOTE is one of the widely used data augmentation methods in sleep stage classification research using electroencephalogram data, which is a long time series signal [91]–[93]. Moreover, heartbeat classification using ECG signal studies used SMOTE to solve data imbalance [94], [95]. We used SMOTE as one of the data augmentation methods of the proposed framework. SMOTE obtained statistically higher performance when SMOTE was compared with ADASYN, ROS, and GAN. Nevertheless, there are still limitations; thus, further studies about effective augmentation methods of ECG data are needed in the future.

In addition, the proposed model requires training time, and the required computational power is not feasible in most clinical settings. Therefore, a model with high computational complexity requires more time to derive results in practical scenarios compared with those measured in a laboratory environment. Therefore, it is necessary to develop a lightweight model in the future.

## VI. CONCLUSION

In this study, we proposed an arrhythmic classification framework that combines ResNet with SE block and biLSTM using the single-lead ECG data. The proposed architecture uses an augmentation method, in particular SMOTE, to solve the problem of imbalance between classification categories. The experimental results showed that the model with augmentation outperformed the model without augmentation on all overall classification metrics in all classification categories. In addition, an ablation study was performed to evaluate the effects of the architecture. The combination of the ResNet, the SE block, and the biLSTM method exhibited an improvement over the ResNet and LSTM methods. Finally, the proposed framework obtained the best generalization ability compared with other deep learning models. Therefore, we confirmed that the ResNet combined with the SE block and the biLSTM architecture classified arrhythmia from single-lead ECG signals without feature extraction.

We could classify multiclass arrhythmia categories, including AFIB, with high accuracy using a model with high classification performance. Because a model trained on data with a large number of classes is fit to a classification type with a large number of classes, the classification performance of a relatively small type is reduced. This problem can be addressed by using SMOTE, which generates synthetic data. Moreover, our proposed model exhibited a high classification performance for the F1-score in minority classes compared to that of the comparison methods using the same database. Finally, our proposed model has the best generalization ability compared to other deep learning models when using few-shot learning and an independent database. Consequently, the proposed model can be useful for long-term ECG monitoring using single-lead wearable devices in clinical settings in the future.

## REFERENCES

[1] T. M. Munger, L.-Q. Wu, and W. K. Shen, "Atrial fibrillation," *J. Biomed. Res.*, vol. 28, no. 1, pp. 1–17, 2014.
[2] S. S. Xu, M.-W. Mak, and C.-C. Cheung, "Towards end-to-end ECG classification with raw signal extraction and deep neural networks," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 4, pp. 1574–1584, Jul. 2019.
[3] S. Khurshid *et al.*, "Frequency of cardiac rhythm abnormalities in a half million adults," *Circulat., Arrhythmia Electrophysiol.*, vol. 11, no. 7, 2018, Art. no. e006273.
[4] J. L. Atlee, *Complications in Anesthesia E-Book*. Amsterdam, The Netherlands: Elsevier, 2006.
[5] L. S. Lilly and E. Braunwald, *Braunwald's Heart Disease: A Textbook of Cardiovascular Medicine*, vol. 2. Amsterdam, The Netherlands: Elsevier, 2012.
[6] S. Thornton and P. Hochachka, "Oxygen and the diving seal," *Undersea Hyperbaric Med.*, vol. 31, no. 1, pp. 81–95, 2004.
[7] A. Galli *et al.*, "Holter monitoring and loop recorders: From research to clinical practice," *Arrhythmia Electrophysiol. Rev.*, vol. 5, no. 2, p. 136, 2016.
[8] G. Sannino and G. De Pietro, "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection," *Future Gener. Comput. Syst.*, vol. 86, pp. 446–455, Sep. 2018.
[9] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, "Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network," *Inf. Sci.*, vol. 405, no. 1, pp. 81–90, Sep. 2017.

[10] D.-O. Won, B.-R. Lee, K.-S. Seo, H. J. Kim, and S.-W. Lee, "Alteration of coupling between brain and heart induced by sedation with propofol and midazolam," *PLoS ONE*, vol. 14, no. 7, Jul. 2019, Art. no. e0219238.

[11] S. Parvaneh, J. Rubin, S. Babaeizadeh, and M. Xu-Wilson, "Cardiac arrhythmia detection using deep learning: A review," *J. Electrocardiol.*, vol. 57, pp. S70–S74, Nov. 2019.

[12] Y. Miyasaka *et al.*, "Secular trends in incidence of atrial fibrillation in Olmsted County, Minnesota, 1980 to 2000, and implications on the projections for future prevalence," *Circulation*, vol. 114, no. 11, pp. E498–E498, 2006.

[13] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," 2017, *arXiv:1707.01836*.

[14] F. Andreotti, O. Carr, M. A. F. Pimentel, A. Mahdi, and M. De Vos, "Comparing feature based classifiers and convolutional neural networks to detect arrhythmia from short segments of ECG," in *Proc. Comput. Cardiol. Conf. (CinC)*, Sep. 2017, pp. 1–4.

[15] V. Sujadevi, K. Soman, and R. Vinayakumar, "Real-time detection of atrial fibrillation from short time single lead ECG traces using recurrent neural networks," in *Proc. Int. Symp. Intell. Syst. Technol. Appl.* Springer, 2017, pp. 212–221.

[16] P. Pławiak, "Novel genetic ensembles of classifiers applied to myocardium dysfunction recognition based on ECG signals," *Swarm Evol. Comput.*, vol. 39, pp. 192–208, Apr. 2018.

[17] Ö. Yıldırım, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Comput. Biol. Med.*, vol. 102, no. 1, pp. 411–420, Nov. 2018.

[18] O. Faust, A. Shenfield, M. Kareem, T. R. San, H. Fujita, and U. R. Acharya, "Automated detection of atrial fibrillation using long short-term memory network with RR interval signals," *Comput. Biol. Med.*, vol. 102, pp. 327–335, Nov. 2018.

[19] C. Chen, Z. Hua, R. Zhang, G. Liu, and W. Wen, "Automated arrhythmia classification based on a combination network of CNN and LSTM," *Biomed. Signal Process. Control*, vol. 57, Mar. 2020, Art. no. 101819.

[20] Y. Liang, S. Yin, Q. Tang, Z. Zheng, M. Elgendi, and Z. Chen, "Deep learning algorithm classifies heartbeat events based on electrocardiogram signals," *Frontiers Physiol.*, vol. 11, p. 1255, Oct. 2020.

[21] R. He *et al.*, "Automatic cardiac arrhythmia classification using combination of deep residual network and bidirectional LSTM," *IEEE Access*, vol. 7, pp. 102119–102135, 2019.

[22] O. Yildirim, M. Talo, E. J. Ciaccio, R. S. Tan, and U. R. Acharya, "Accurate deep neural network model to detect cardiac arrhythmia on more than 10,000 individual subject ECG records," *Comput. Methods Programs Biomed.*, vol. 197, Dec. 2020, Art. no. 105740.

[23] A. M. Shaker, M. Tantawi, H. A. Shedeed, and M. F. Tolba, "Generalization of convolutional neural networks for ECG classification using generative adversarial networks," *IEEE Access*, vol. 8, pp. 35592–35605, 2020.

[24] F. Murat *et al.*, "Exploring deep features and ECG attributes to detect cardiac rhythm classes," *Knowl.-Based Syst.*, vol. 232, Nov. 2021, Art. no. 107473.

[25] M. Chen *et al.*, "Region aggregation network: Improving convolutional neural network for ECG characteristic detection," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 2559–2562.

[26] S. Banerjee and M. Mitra, "Application of cross wavelet transform for ECG pattern analysis and classification," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 326–333, Feb. 2014.

[27] T. M. Khoshgoftaar, C. Seiffert, J. V. Hulse, A. Napolitano, and A. Folleco, "Learning with limited minority class data," in *Proc. 6th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2007, pp. 348–353.

[28] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *J. Big Data*, vol. 5, no. 1, pp. 1–30, Dec. 2018.

[29] F. Thabtah, "An accessible and efficient autism screening method for behavioural data and predictive analyses," *Health Inform. J.*, vol. 25, no. 4, pp. 1739–1755, 2019.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[32] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[33] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, Jan. 2002.

[34] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, May/Jun. 2001.

[35] G. Clifford *et al.*, "AF classification from a short single lead ECG recording: The PhysioNet computing in cardiology challenge 2017," in *Proc. Comput. Cardiology Conf.*, Sep. 2017, pp. 1–4.

[36] L. Xu, M. Xu, Y. Ke, X. An, S. Liu, and D. Ming, "Cross-dataset variability problem in EEG decoding with deep learning," *Frontiers Hum. Neurosci.*, vol. 14, p. 103, Apr. 2020.

[37] M. Lee, J.-H. Jeong, Y.-H. Kim, and S.-W. Lee, "Decoding finger tapping with the affected hand in chronic stroke patients during motor imagery and execution," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1099–1109, 2021.

[38] Q. Ye, B. Y. Lin, and X. Ren, "CrossFit: A few-shot learning challenge for cross-task generalization in NLP," 2021, *arXiv:2104.08835*.

[39] J.-H. Jeong, K.-H. Shim, D.-J. Kim, and S.-W. Lee, "Brain-controlled robotic arm system based on multi-directional CNN-BiLSTM network using EEG signals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 5, pp. 1226–1238, May 2020.

[40] T. Liu, Y. Yang, W. Fan, and C. Wu, "Few-shot learning for cardiac arrhythmia detection based on electrocardiogram data from wearable devices," *Digit. Signal Process.*, vol. 116, Sep. 2021, Art. no. 103094.

[41] Z. Tian *et al.*, "Generalized few-shot semantic segmentation," 2020, *arXiv:2010.05210*.

[42] L. Xu, M. Xu, Z. Ma, K. Wang, T.-P. Jung, and D. Ming, "Enhancing transfer performance across datasets for brain-computer interfaces using a combination of alignment strategies and adaptive batch normalization," *J. Neural Eng.*, vol. 18, no. 4, 2021, Art. no. 0460e5.

[43] M. Völker, R. T. Schirrmeister, L. D. J. Fiederer, W. Burgard, and T. Ball, "Deep transfer learning for error decoding from non-invasive EEG," in *Proc. 6th Int. Conf. Brain-Comput. Interface (BCI)*, Jan. 2018, pp. 1–6.

[44] M. Lee *et al.*, "Quantifying arousal and awareness in altered states of consciousness using interpretable deep learning," *Nature Commun.*, vol. 13, no. 1, pp. 1–14, Dec. 2022.

[45] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Jan. 2002.

[46] P. Branco, L. Torgo, and R. Ribeiro, "A survey of predictive modelling under imbalanced distributions," 2015, *arXiv:1505.01658*.

[47] V. Ganganwar, "An overview of classification algorithms for imbalanced datasets," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 2, no. 4, pp. 42–47, Apr. 2012.

[48] A. Liu, J. Ghosh, and C. E. Martin, "Generative oversampling for mining imbalanced datasets," in *Proc. Int. Conf. Data Mining*, 2007, pp. 66–72.

[49] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[50] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 4, pp. 588–598, Sep. 2017.

[51] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jun. 2008, pp. 1322–1328.

[52] J. Brandt and E. Lanzén, "A comparative review of SMOTE and ADASYN in imbalanced data classification," DIVA, Thane, Maharashtra, Tech. Rep. 1 519 153, 2021, pp. 1–42.

[53] M. Khalifa and K. Shaalan, "Character convolutions for Arabic named entity recognition with long short-term memory networks," *Comput. Speech Lang.*, vol. 58, pp. 335–346, Nov. 2019.

[54] J. Zheng, J. Zhang, S. Danioko, H. Yao, H. Guo, and C. Rakovski, "A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients," *Sci. Data*, vol. 7, no. 1, pp. 1–8, Dec. 2020.

[55] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of imbalanced data: Review of methods and applications," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 1099, no. 1, 2021, Art. no. 012077.

[56] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, 2016.

[57] A. Saifudin and R. S. Wahono, "Pendekatan level data untuk menangani ketidakseimbangan kelas pada prediksi cacat software," *J. Softw. Eng.*, vol. 1, no. 2, pp. 76–85, 2015.

[58] S. Maldonado, J. López, and C. Vairetti, "An alternative SMOTE oversampling strategy for high-dimensional datasets," *Appl. Soft Comput.*, vol. 76, pp. 380–389, Mar. 2019.

[59] A. G. Pertiwi, N. Bachtiar, R. Kusumaningrum, I. Waspada, and A. Wibowo, "Comparison of performance of $k$-nearest neighbor algorithm using smote and $k$-nearest neighbor algorithm without smote in diagnosis of diabetes disease in balanced data," *J. Phys., Conf. Ser.*, vol. 1524, no. 1, Apr. 2020, Art. no. 012048.

[60] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 4, pp. 1119–1130, Aug. 2012.

[61] P. Skryjomski and B. Krawczyk, "Influence of minority class instance types on SMOTE imbalanced data oversampling," in *Proc. Int. Workshop Learn. Imbalanced Domains, Theory Appl.*, 2017, pp. 7–21.

[62] M. Koziarski, "Two-stage resampling for convolutional neural network training in the imbalanced colorectal cancer image classification," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.

[63] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[65] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[66] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," 2015, *arXiv:1505.00853*.

[67] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *Proc. Int. Conf. Artif. Intell.*, 2011, pp. 1–6.

[68] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.

[69] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, vol. 26. New York, NY, USA: Springer, 2013.

[70] X. Huang, B. He, M. Tong, D. Wang, and C. He, "Few-shot object detection on remote sensing images via shared attention module and balanced fine-tuning strategy," *Remote Sens.*, vol. 13, no. 19, p. 3816, Sep. 2021.

[71] J.-W. Seo, H.-G. Jung, and S.-W. Lee, "Self-augmentation: Generalizing deep networks to unseen classes for few-shot learning," *Neural Netw.*, vol. 138, pp. 140–149, Jun. 2021.

[72] H.-G. Jung and S.-W. Lee, "Few-shot learning with geometric constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4660–4672, Nov. 2020.

[73] R. T. Schirrmeister *et al.*, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Hum. Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Aug. 2017.

[74] W. Yan and H. Shao, "Application of support vector machine nonlinear classifier to fault diagnoses," in *Proc. 4th World Congr. Intell. Control Autom.*, vol. 4, 2002, pp. 2697–2700.

[75] L. C. D. Nkengfack, D. Tchiotsop, R. Atangana, B. S. Tchinda, V. Louis-Door, and D. Wolf, "A comparison study of polynomial-based PCA, KPCA, LDA and GDA feature extraction methods for epileptic and eye states EEG signals detection using kernel machines," *Informat. Med. Unlocked*, vol. 26, Jan. 2021, Art. no. 100721.

[76] E. J. D. S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti, "ECG-based heartbeat classification for arrhythmia detection: A survey," *Comput. Methods Programs Biomed.*, vol. 127, pp. 144–164, Apr. 2016.

[77] M. A. Khan and Y. Kim, "Cardiac arrhythmia disease classification using LSTM deep learning approach," *Comput., Mater. Continua*, vol. 67, no. 1, pp. 427–443, 2021.

[78] J.-H. Cho, J.-H. Jeong, and S.-W. Lee, "NeuroGrasp: Real-time EEG classification of high-level motor imagery tasks using a dual-stage deep learning framework," *IEEE Trans. Cybern.*, early access, Nov. 8, 2021, doi: 10.1109/TCYB.2021.3122969.

[79] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Mach. Learn.*, vol. 30, nos. 2–3, pp. 195–215, Feb. 1998.

[80] L. van der Maaten, "Barnes-hut-SNE," 2013, *arXiv:1301.3342*.

[81] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 935–942.

[82] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new oversampling method in imbalanced data sets learning," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, 2005, pp. 878–887.

[83] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased LSTM: Accelerating recurrent network training for long or event-based sequences," 2016, *arXiv:1610.09513*.

[84] F. Zhu, F. Ye, Y. Fu, Q. Liu, and B. Shen, "Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network," *Sci. Rep.*, vol. 9, no. 1, pp. 1–11, Dec. 2019.

[85] Y. Jin, C. Qin, Y. Huang, W. Zhao, and C. Liu, "Multi-domain modeling of atrial fibrillation detection with twin attentional convolutional long short-term memory neural networks," *Knowl.-Based Syst.*, vol. 193, Apr. 2020, Art. no. 105460.

[86] G. Petmezas *et al.*, "Automated atrial fibrillation detection using a hybrid CNN-LSTM network on imbalanced ECG datasets," *Biomed. Signal Process. Control*, vol. 63, Jan. 2021, Art. no. 102194.

[87] U. R. Acharya *et al.*, "A deep convolutional neural network model to classify heartbeats," *Comput. Biol. Med.*, vol. 89, pp. 389–396, Oct. 2017.

[88] E. Combrisson and K. Jerbi, "Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy," *J. Neurosci. Methods*, vol. 250, pp. 126–136, Jul. 2015.

[89] S.-H. Lee, M. Lee, and S.-W. Lee, "Neural decoding of imagined speech and visual imagery as intuitive paradigms for BCI communication," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 12, pp. 2647–2659, Dec. 2020.

[90] M. Lee *et al.*, "Connectivity differences between consciousness and unconsciousness in non-rapid eye movement sleep: A TMS–EEG study," *Sci. Rep.*, vol. 9, no. 1, pp. 1–9, Dec. 2019.

[91] C. Sun, J. Fan, C. Chen, W. Li, and W. Chen, "A two-stage neural network for sleep stage classification based on feature learning, sequence learning, and data augmentation," *IEEE Access*, vol. 7, pp. 109386–109397, 2019.

[92] S. Mousavi, F. Afghah, and U. R. Acharya, "SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach," *PLoS ONE*, vol. 14, no. 5, May 2019, Art. no. e0216456.

[93] A. Salamatian and A. Khadem, "Automatic sleep stage classification using 1D convolutional neural network," *Frontiers Biomed. Technol.*, vol. 7, no. 3, pp. 142–150, Nov. 2020.

[94] S. K. Pandey and R. R. Janghel, "Automatic detection of arrhythmia from imbalanced ECG database using CNN model with SMOTE," *Australas. Phys. Eng. Sci. Med.*, vol. 42, no. 4, pp. 1129–1139, Dec. 2019.

[95] S. Mousavi and F. Afghah, "Inter- and intra-patient ECG heartbeat classification for arrhythmia detection: A sequence to sequence deep learning approach," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1308–1312.

**Yun Kwan Kim** received the B.S. degree in occupational therapy from Dongnam Health University, Suwon, South Korea, in 2012, and the M.S. degree in cognitive science from Sungkyunkwan University, Seoul, South Korea, in 2016. He is currently pursuing the Ph.D. degree with the Department of Brain and Cognitive Engineering, Korea University, Seoul.

He is currently a Senior Researcher with SEERS Technology Company Ltd., Seongnam-si, Republic of Korea, responsible for the development of a machine-learning-based algorithm using biomedical data. His research interests include artificial intelligence, biomedical informatics, and neuroscience.

**Minji Lee** received the B.S. degree in computer education and biological sciences and the M.S. degree in health sciences and technology from Sungkyunkwan University, Seoul, South Korea, in 2009 and 2015, respectively, and the Ph.D. degree from the Department of Brain and Cognitive Engineering, Korea University, Seoul, in 2021.

She is currently a Data Scientist with SK Hynix, Icheon-si, South Korea. Her research interests include machine learning, artificial intelligence, and human–computer interaction.

**Hee Seok Song** received the B.S. and M.S. degrees in electronics engineering from Sogang University, Seoul, South Korea, in 1997 and 2000, respectively.

He is currently the Vice-President and the Chief Technical Officer of SEERS Technology Company Ltd., Seongnam-si, Republic of Korea. His research interests include remote patient monitoring systems using wearable medical devices and biosignal analysis algorithms.

**Seong-Whan Lee** (Fellow, IEEE) received the B.S. degree in computer science and statistics from Seoul National University, Seoul, South Korea, in 1984, and the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 1986 and 1989, respectively.

He is currently the Head of the Department of Artificial Intelligence, Korea University, Seoul, South Korea. His research interests include artificial intelligence, pattern recognition, and brain engineering.

Dr. Lee is also a fellow of the International Association of Pattern Recognition (IAPR) and the Korea Academy of Science and Technology.