IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 14, Issue 3 (May. - June. 2024), pp.14-25
e-ISSN: 2319-4200, p-ISSN: 2319-4197
www.iosrjournals.org

Contrast-CLGAN: Contrastive Continual Learning using GAN-based Data Generation for ECG based Heart Diseases Detection System

Nassim Ammour
Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Abstract
Irregular heartbeats due to abnormal electrical heart activity are symptom of Cardiovascular disease (CVD), it is a source of stroke, blood clots, heart failure and other heart-related complications. Most of the developed Electrocardiogram (ECG) based automatic cardiac arrhythmia detection systems require the availability of a large data with all arrhythmias for the training process, and cannot be updated without adequate data and cost. Therefore, this paper aims to develop a continual learning method by introducing incrementally new arrhythmias to a deep learning CVD detection system already trained with old ones. However, due to the catastrophic forgetting phenomenon, the pre-trained model loses its pre-acquired knowledge and performs poorly, if it is subject to a new training process. To overcome this handicap, we propose a new model neural architecture Contrast-CLGAN consisting of two trainable submodules. The first module adjusts its weights in such a way as to classify the input Electrocardiogram signal and distinguish between the arrhythmias, using cross-entropy and mean square error loss functions. Simultaneously, the second module tries to keep a trace of the seen data and memorizes them by learning its parameters to discover, extract, and retain the hidden structure of the training data. A contrastive learning module is intercalated before the classifying module to enhance the distinction between the different classes and, therefore, boost the accuracy of the deep model. By comparing the proposed method to the state-of-the-art, the proposed model showed improved performance in classifying different arrhythmias from the MIT-BIH, INCART, and SVDB databases.
Keywords*: Electrocardiogram classification; continual learning; catastrophic forgetting; generative model, contrastive learning.*

I. Introduction and literature review

The heart is the main organ of the human cardiovascular system that pumps incessantly blood throughout the body during the whole lifespan. The heart walls (three layers, inner layer called the Endocardium, the middle layer called the Myocardium, and the outer layer called the Epicardium) are the muscles that contract and relax to move and transmit blood throughout the body. The heart contains four main sections, two upper chambers (the atria) and two lower chambers (the ventricles), made up of muscle and powered by electrical impulses. The role of these chambers is vital in the process of pumping blood throughout the entire body via the vessels of the circulatory system, maintaining the right amount of dissolved oxygen in the blood, and controlling the blood pressure[1].

CVDs, which are conditions that affect the structures or function of the heart, are a serious societal problem. Every year, 17.3 million persons die from cardiovascular disease [2]. CVDs are the most leading cause of death, constituting over 31% of deaths around the world [3]. More than 50% of sudden cardiac deaths result from cardiac arrhythmias and are causing half of the deaths on account of all heart diseases [4]. The simple and effective option to diagnose cardiac arrhythmias is ECG. The latter is a non-invasive and easy-to-use way to obtain abundant health and pathology information about the heart [5]. Electrocardiography reveals the pathological states of cardiovascular systems by alterations in their wave shapes or rhythms [6]. Often, the randomness of arrhythmia events and differences in doctors' skills cause misdiagnosis in clinical practice. Accurate automatic cardiac arrhythmia events detection and identification help doctors monitor heart conditions.

Cardiac monitoring and heart abnormalities detection using an automatic ECG signals classification system is a challenging task. In recent years, several works for cardiac arrhythmia detection have been conducted[1]. In [7]–[9], cardiac arrhythmias are detected using wavelet features and independent component. Morphological features extraction techniques used to build an automatic ECG signal classification system are developed in [10]–[12]. In [13], [14] , authors used a support machine vector (SVM) to classify heartbeat signal samples. Higher-order statistic feature extraction technique is used in [15], [16] for an efficient ECG signal classification. Artificial Neural Network (ANN) architecture derives from the ability to create a computer system that simulate the human brain. ANNs are composed of three main (input, hidden and output) layers. The connections are set up between layers through weights and bias. ANN are implemented to learn and approximate complex nonlinear function [17], [18]. ANN based techniques have been used intensively and effectively to classify ECG signals in [19], [20].

Signal processing models and analyses data representations of measured physical events. Deep learning, which has aroused as an effective tool for analysing big data, gives new opportunities to generate predictive models to solve a large variety of signal processing applications. Within Deep Learning, a Convolutional Neural Network (CNN), which is a type of artificial neural network inspired by the architecture of the human brain, is widely used for pattern recognition and classification [21], [22]. In the last few years, CNN models have been used effectively and efficiently to detect heartbeat arrhythmias. Feature extraction is a crucial process before classification. CNN is a sequence of convolution layers, activation function, pool layer, and fully-connected layer linked to extract optimal discriminative features combined with a classifier to diagnose cardiac disease [18], [19]. Most of the existing ECG classification methods, including the CNN-based deep learning models, deem that all arrhythmia classes are well known during the initial learning phase. Consequently, providing a dataset containing all the classes is required for the learning phase. Humans have the potential to continuously acquire, fine-tune, and transfer skills and knowledge throughout their lifetime. This paper addresses the issue of learning various sequential learning tasks, where during each new learning task, there are a set of new arrhythmia classes to learn. Suppose that the old learning task consists of classifying an ECG dataset labelled into N classes. Subsequently, a new task consisting of the classification of a set of M classes ECG dataset is added. The most straightforward way is to amalgamate the two ECG datasets and train a deep model to classify ECG heartbeats into N+M classes. Nevertheless, it may be hard to maintain a good classification performance due to the lack of an old task dataset when dealing with the new task. Moreover, if we retrain the pre-trained deep model on the new M classes dataset, the retrained model will mislay its high classification performance for the first dataset. Retraining the model on a new classification task will disturb its parameters, and consequently, the model will forget about the old classification task and perform weakly. This performance deterioration phenomenon is known as 'catastrophic forgetting'[25]–[27]. This situation can also be deemed as a multi-task classification problem, where the classification using different datasets at different times is considered a disjoint classification task. In this case, the we need the availability of all the datasets during the training phase. In the machine-learning community, this problem is well known as learning-without-forgetting (LwF) [28], incremental learning, or continual learning [29].

Several techniques for incremental learning are proposed in the deep learning community. In[30], the authors proposed to extend a deep neural network model by adding convolutional layers associated with the unseen classes of the new task. In [31], the authors proposed to fight the forgetting phenomena in the continual learning process by adding new nodes and augmenting the layer size in the network using a controller network. Elastic weight consolidation (ECW) is proposed in [32], the authors proposed to reduce the learning effect selectively on significant weights for the old tasks using regularization parameters. Finally, the authors in[33] proposed a solution (Lwf-ECG) based on deep model that combines pre-trained CNN for feature extraction, a memory module to store one prototype for each task, and a deep model for task selection to classify an ECG heartbeat signals.

Specifically, in this research paper, we propose a new trainable deep learning model able to perform in a continual learning manner without degrading when executing old classification tasks. The proposed deep neural network architecture includes three trainable modules. The first module is an EfficientNet based feature extraction model. The second module is a contrastive learning model followed by a classification model. Normalized features from the same class are pulled closer together, and features from different classes are pushed away from each other. The third module is a Generative Adversarial Networks (GAN), used as a generative model to memorize the latent structure of the old tasks datasets.

II. Materials and Methods

Given a long series of tasks $T_k = \left\{X_i^{(k)}, y_i^{(k)}\right\}_{i=1}^{N_k}, k = 1, \ldots, K$, where during each ECG heartbeat classification task T_k, we aim to train the same previously trained deep model on the new subset c_k classes with N_k ECG data $X^{(k)}$ and their corresponding categorical class labels $y^{(k)}$, without forgetting the old tasks classification performance. For this sake, we propose a new deep learning architecture composed of three main modules, as illustrated in Figure 1. The first module contains a feature extraction backbone CNN responsible for producing good features to ensure effective discrimination between the ECG classes of both new and old tasks. The role of the second module is to memorize the hidden structure of the old tasks dataset, and it is trained to generate data representing the already seen classes. The third module is a deep contrastive optimization architecture used to increase the disparity between the ECG heartbeat classes and boost the model performance. Next, we will discuss the different parts that constitute the proposed architecture.

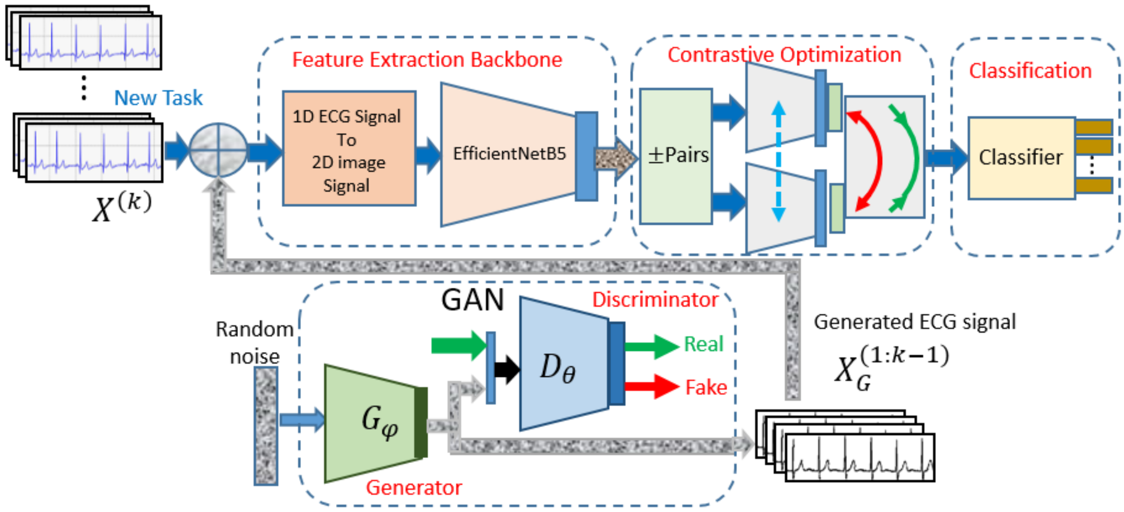


Figure1. Contrastive learning and GAN based hybrid model.

2.1 Feature extraction module

Feature extraction is the most crucial step of biomedical signal processing. The goal of feature extraction is to find the most compacted and informative set of features to enhance the discrepancy between the ECG data classes. For images data, pre-trained CNNs are very effective feature extraction modules. To benefit from this remark, which appeared in the literature [34], we converted the 1D ECG signal into an image before the feature extraction phase. In [34], the conversion of the 1D ECG signal to a 2D image is obtained via training a series of CNN layers.

The feature extraction block contains two sub-networks as illustrated in Figure 2. The role of the first sub-network is to convert the 1D ECG heartbeat into an image. The ECG signal is fed into a fully-connected layer with dimension 1024 followed by fully-connected, reshaping, up-sampling, and convolution layers to produce an image of dimension (3,28,28). The second sub-network is an EfficientNetB5 used to extract compacted and informative features from the ECG images. EfficientNetB5 is a recent pre-trained CNN that has illustrated remarkable performances in image classification. EfficieNeTB5 is a CNN deep architecture using a scaling method that uniformly scales the dimensions of depth/width/resolution using compound coefficient [35]. The top softmax classification layer of the EfficientNetB5 is removed and replaced with a fully connected layer followed by a reshape layer to extract an ECG feature of dimension (1,20,20). This extracted ECG heartbeats grayscale image size is chosen in order to reduce the computation time, but an image of different sizes can be chosen.

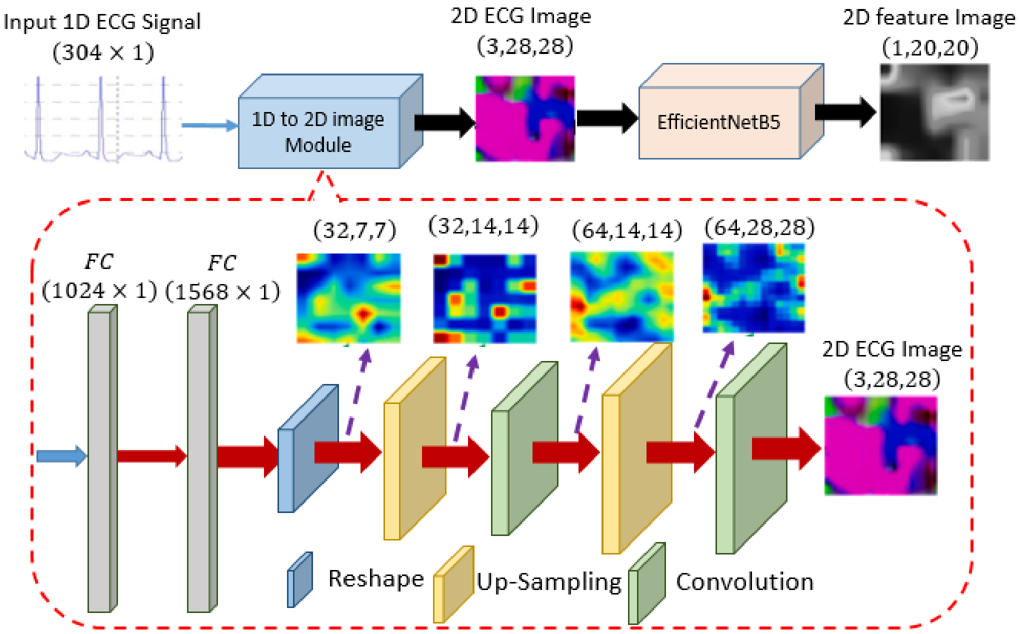


Figure 2. Feature extraction module and 1D ECG signal to image CNN conversion

2.2 Old tasks' features generation

In order to memorize the latent structure of the previously seen data, a deep learning-based generative model is used to control the forgetting phenomena. A generative adversarial network, known as GAN for short, is used to learn the complex latent space of the old tasks' data and try to reproduce data with the same structure or clusters. The architecture of the GAN is composed of a generator $G_{\varphi}(\cdot)$ with learnable parameters φ and a discriminator $D_{\theta}(\cdot)$ with learnable parameters θ as shown in Figure 1. The generator is trained to fake the discriminator by reproducing the real heartbeat ECG signal from noise, and the discriminator is trained to distinguish between the generated/fake signal and the real signal. The discriminator model and the generator model play the minimax game with value function given in Equation 1:

$$\mathcal{L}_{GAN} = \min_{G_{\varphi}} \max_{D_{\theta}} \left[\mathbb{E}_{x\sim p_{data}(x)}[\log D_{\theta}(x)] + \mathbb{E}_{x\sim p_{z}(z)}\left[\log\left(1 - D_{\theta}(G_{\varphi}(z))\right)\right]\right] \quad (1)$$

For a mini-batch of m noise samples $\{z^{(1)}, \ldots, z^{(m)}\}$ from noise prior $P_g(z)$ and a mini-batch of m samples $\{x^{(1)}, \ldots, x^{(m)}\}$ from data generation distribution $P_{data}(x)$, the discriminator is updated by ascending its stochastic gradient using Equation 2,

$$\nabla_{\theta} \frac{1}{m} \sum_{i=1}^{m} \left[\log D_{\theta}\left(x^{(i)}\right) + \log\left(1 - D_{\theta}\left(G_{\varphi}\left(z^{(i)}\right)\right)\right)\right] \quad (2)$$

For a mini-batch of m noise samples $\{z^{(1)}, \ldots, z^{(m)}\}$ from noise prior $P_g(z)$, the generator is updated by descending its stochastic gradient given in Equation 3,

$$\nabla_{\varphi} \frac{1}{m} \sum_{i=1}^{m} \left[\log\left(1 - D_{\theta}\left(G_{\varphi}\left(z^{(i)}\right)\right)\right)\right] \quad (3)$$

2.3 Contrastive learning

Recently, the contrastive learning technique has led to significant enhancements in self-supervised representation learning. A contrastive optimization module is added before the classification process to increase the disparity between ECG heartbeat signal classes and boost the classifier performance. The key idea in the contrastive learning strategy is to create pairs of positive samples, the anchor with a sample from the same class that are attracted in the embedding space, and pairs of negative samples, an anchor with sample from different class that are repealed in embedding space as illustrated in Figure 3. The concept in the contrastive learning strategy is to create pairs of positive samples taken from the same class and negative samples taken from the rest of the classes. The anchor attracts the positive sample and repeals the negative sample in the embedding space.

The contrastive optimization module encapsulates an encoder network $Enc(\cdot)$ to map the vector $p = ftr(x) \in \mathcal{R}^{20\times20\times1}$(the feature of the input sample x) into a normalized vector $r = Enc(p) \in \mathcal{R}^{2048}$, and a projection network $Proj(\cdot)$ to map the output of the encoder to a vector $q = Proj(r)$an embedding into a normalized unit hypersphere in $\mathcal{R}^{128}$. The projection network, discarded at the inference time, is used only in the contrastive training step.

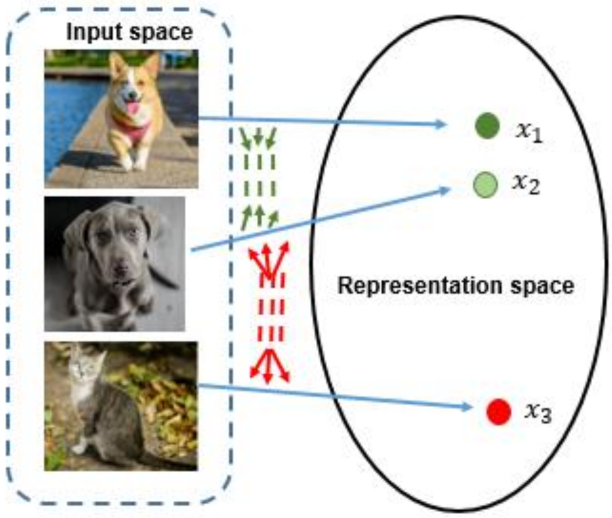


Figure 3. Contrastive learning based representation space

For a pair of ECG heartbeat data $\{x_i, x_j\}$ and their representations $\{q_i, q_j\}$, the distance between the representations of the input data x_i and x_j in the latent space is computed using Equation 4:

$$d(x_i, x_j) = \|q - q\|_2 = \left\|Enc(Proj(p_i)) - Enc\left(Proj(p_j)\right)\right\|_2 \quad (4)$$

By using Equation 5, we can obtain the contrastive loss function $\mathcal{L}_{CL}$ as:

$$\mathcal{L}_{CL}(x_i, x_j) = 0.5\left[y(x_i, x_j) \cdot d(x_i, x_j)^2 + (1 - y(x_i, x_j)) \cdot max(m - d(x_i, x_j), 0)^2\right] + \lambda\|w\|^2 \quad (5)$$

Where the binary ground truth label $y(x_i, x_j) = 0$ indicates that the pair of images are not-similar (anchor with negative sample), and $y(x_i, x_j) = 1$ for similar pair images (pair of samples from the same class). The hyper-parameter m is the margin representing the threshold for the non-similar pair of images chosen by the user.The hyper-parameter λ controls the regularization term. The contrastive objective function $\mathcal{L}_{CL}$is minimized to reach the optimal parameters of the encoder network$Enc(\cdot)$ and the projection network $Proj(\cdot)$. Thus, the feature samples from the same class are attracted to be closer to each other in the embedding space, and thefeature samples from different classes are repelled and brought far from each other.

2.4 Proposed continual learning algorithm

In the continual learning process, the classification model performs a sequence of K classification tasks, and for each task, only the new unseen classes are available (i.e. the data for the old and future tasks is not accessible). The objective in the continual learning process is to learn the new task and perform well on all the already seen tasks. To implement a sequential learning process, the ECG data set is split into K tasks, each classification task T_k includes N_k ECG heartbeats $X^{(k)}$ with their corresponding categorical class labels $Y^{(k)}$ representing c_k classes. For instance, the MIT-BIH dataset contains 12 classes, thus we can build a continual learning process using 6 tasks with 2 classes per task.

After $k - 1$ tasks, our target is to train the model on a new task T_k. First, the ECG heartbeat signals $X^{(k)}$ of task T_k are converted to ECG images $I^{(k)}$. Then, features from $I^{(k)}$ are extracted using the EfficientB5 backbone network $W^{(k)} = ftr(I^{(k)})$. To fight the forgetting phenomena, a GAN architecture is used to generate the old tasks heartbeat signal$\left(X_G^{(1)}, \dots, X_G^{(k-1)}\right)$ (the GAN was trained on the task T_{k-1}heartbeat signals gathered with the generated heartbeat signals of the old tasks $(T_1, \dots, T_{k-2})$). Afterward, a contrastive learning-based module is trained using positive and negative feature pairs, randomly selected from the features dataset $\left(W^{(1)}, \dots, W^{(k)}\right)$, to enhance the disparity between the data clusters and boost the classifier performance. Finally, a classifier is trained to classify the enhanced features into c_k classes.The architecture of the proposed continual learning model is illustrated in Figure 1. During each new taskT_k, the parameters of the feature extraction backbone and the classifier are learned by training and optimizing the loss functions of the model using the training subset of the actual classification task dataset. The training dataset of the actual task is created using the new task T_k training dataset (new unseen classes) gathered with the generated synthetic dataset of the old classification tasks $T_{k-1}, T_{k-2}, \dots, T_1$.In parallel, the parameters of the GAN model are learned by training the GAN on the same training dataset of the actual classification task. Finally, we furnish a procedural description of the proposed algorithm Contrast-CLGAN in algorithm 1.

Algorithm 1:

Start with:

θ_{ftr}:parameters of the feature extraction module.
θ_{CL}:parameters of the contrastive learning module.
$\theta_{classif}$:parameters of the classificationlayer.
θ_{GAN}:parameters of the feature generation module.
$\{X^{(k)}, Y^{(k)}\}$:training data and ground truth of the new task.

Initialize:

$z = \{z_i\}_{i=1}^{M} \leftarrow \mathcal{N}(0,1)$ // generate M random noise vectors.
$X_G^{(1:k-1)} = \{X_G^{(i)}\}_{i=1}^{M} \leftarrow G_\varphi(z)$ // generate the old tasks' features from the trained GAN's generator.
$Y_G^{(1:k-1)} \leftarrow CNN_{classif}(\hat{X}_G^{(1:k-1)})$ // get the labels of the generated features.
$\theta_{ftr}, \theta_{CL}, \theta_{GAN} \leftarrow RandInit(|\theta_{ftr}|, |\theta_{CL}|, |\theta_{GAN}|)$// randomly initialize the model parameters.

Train:

Define $\hat{Y}^k \leftarrow CNN_{classif}(X^{(k)}, \theta_{ftr}, \theta_{CL}, \theta_{classif})$ // New task output

$$\theta_{ftr}^*, \theta_{CL}^*, \theta_{classif}^* \leftarrow \underset{\theta_{ftr}, \theta_{CL}, \theta_{classif}}{\text{argmin}} \left(\mathcal{L}_{classif}(\hat{Y}^{(k)}, Y^{(k)}) + \mathcal{L}_{CL}(w^{(l)}, w^{(n)})\right)$$

Define $X_G^{(1:k-1)} \leftarrow CNN_{GAN}(z, \theta_{GAN})$ // GAN model output

$$\theta_{GAN}^* \leftarrow \underset{\theta_{GAN}}{\text{argmin}} \left(\mathcal{L}_{GAN}([X_G^{(1:k-1)}, X^{(k)}], z)\right)$$

III. Experimental results and discussion

Experiments are conducted on three known ECG datasets to evaluate the proposed Contrast-CLGAN method. Namely, the MIT-BIH dataset[36], the INCART dataset[37], and the SVDB dataset[38]. The results of the experiments are compared with the recent works in the literature to illustrate the proposed method's performances.

3.1 Dataset description

Three ECG database are used to show the capabilities of the Contrast-CLGAN model, as illustrated in Table 1. The detailed class statistics for these databases are given in Table 2. Figure 4 illustrates some samples of the used ECG signals. Before performing the experiments, the ECG signals are segmented into heartbeats using Ecgpuwave software and resampled to a fixed size of 300 samples [39], [40].

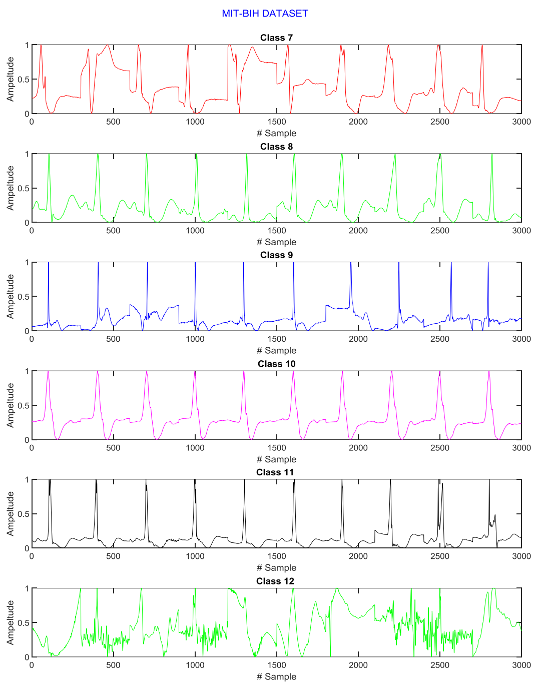
Table 1. ECG datasets that are used in the experiments.

Dataset	Field	Value
MIT BIH	Description:	MIT-BIH Arrhythmia Database contains 48 records with 30 min duration each of two-channel ambulatory ECG recordings obtained from 47 subjects. The recordings were digitized at 360 Hz per channel with an 11-bit resolution.
	No. of records:	48 records, each record is 30 min in length
	Sample rate (Hz)	360
	No. of leads	2
INCART	Description:	INCART database contains 75 annotated recordings extracted from 32 Holter records. Each record is built of a half-hour ECG recording sampled at a rate of 257 Hz and contains 12 standard leads.
	No. of records:	75 records. Each record is 30 min in length.
	Sample rate (Hz)	257
	No. of leads:	12
SVDB	Description:	This database consists of 78 two-lead recordings with a duration of approximately 30 min and sampled at a rate of 128 Hz.
	No. of records:	78 records. Each record is 30 min in length
	Sample rate (Hz)	128
	No. of leads:	2

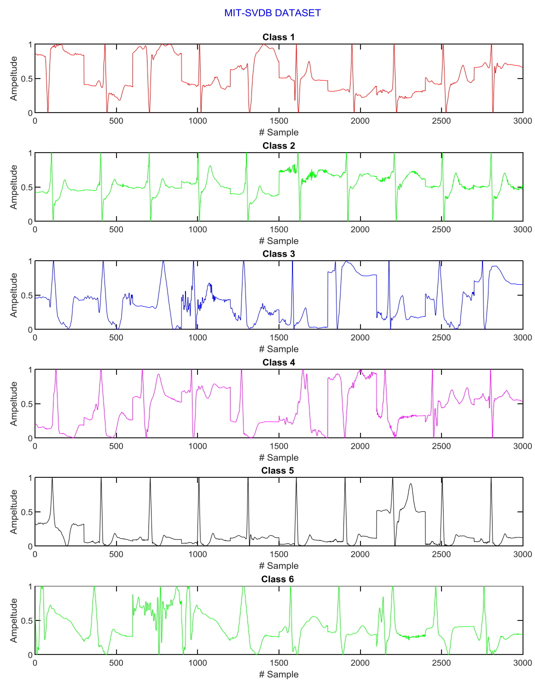
Table 2. Class names and distributions of used ECG datasets.

Class name	Class description	Dataset		
		MIT-BIH	MIT-SVDB	INCART

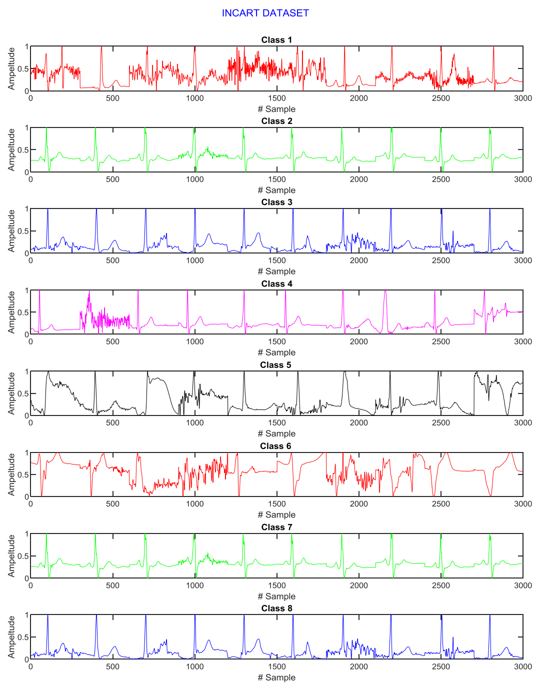
N	Normal beat	76820	145436	150283
L	Left bundle branch block beat	8069	-	-
R	Right bundle branch block beat	9414	-	3170
A	Atrial premature beat	2647	-	1942
a	Aberrated atrial premature beat	150	-	-
J	Nodal (junctional) premature beat	83	9	-
s	Supraventricular premature beat	-	-	16
V	Premature ventricular contraction	6987	10723	20000
F	Fusion of ventricular and normal beat	803	8281	219
j	Nodal (junctional) escape beat	229	23	92
E	Ventricular escape beat	106	-	-
f	Fusion of paced and normal beat	722	-	-
Q	Normal beat	33	74	6
Total		106,063	175,728	164,546



(a)



(b)



(c)

Figure 4. ECG signal record samples: (a) MIT-BIH database, (b) SVDB database, and (c) INCART database.

3.2 Experiment setup and performance evaluation

To test the Contrast-CLGAN model and perform a continual learning process, we divided each dataset into a group of tasks. For each new task, 80% from the new task dataset is used for the training, and 20% is saved with the old classification tasks testing sets to be used in the testing time.

To preserve the acquired knowledge about the old classification tasks, a GAN model, which comprises a generator (to generate synthetic samples) and a discriminator (to discriminate between the true samples and the fake generated samples), is used to learn the unknown probability distribution of the dataset. The generator architecturehasan input layer of shape100 nodes (noise), three hidden layers $(256, 512, 1024)$ nodes with the Leaky-Relu activation function, each hidden layer is followed by a batch normalization layer, and an output layer of shape $(20 \times 20 \times 1)$.The discriminator architecture comprises an input layer of shape $(20 \times 20 \times 1)$, two hidden layers (with Leaky-Relu activation function) of shape 512 and 256 respectively, and an output layer of one node. The GAN has $592{,}897$trainable parameters to be learn.

The old task features, generated using the trained GAN, are added to the new task features to train the contrastive learning module (two tower modules using an embedding module with an input layer$(20 \times 20 \times 1)$, Batch-Normalization layer, Flatten layer, hidden layer $(256\, nodes)$, the output is a one node dense layer) and the classifier (Multilayer Perceptron MLP with input $(20 \times 20 \times 1)$, hidden layer (1024) , and an output layer with number of classes outputs). Adam optimizer with learning rate of 0.002 is used to train the GAN. Stochastic gradient descent (SGD) optimization algorithm with a learning rate of 0.01 is used to learn the parameters of the classifier and a contrastive loss with margin $m = 1$and RMSprop optimizer are used to train the contrastive learning model. To perform the task T_k, we train the total model (feature extraction backbone, the classifier with contrastive optimization, and the GAN) on the k^{th} task dataset. Then, we generate synthetic features of task T_{k-1} using the GAN and add them to the new task dataset features to train the classifier and the GAN for the new task T_k. All the experiments were conducted on the Google Colaboratory cloud service using the available GPU to accelerate the deep learning process.

3.3 Experimental results

In the first experiment, and before performing the continual learning process, the Contrast-CLGAN model was experimented on the total dataset using all the classes. An accuracy of 99.04%, 98.9%, and 97.74%was obtained for the MITBIH, INCART, and SVDB datasets respectively. Afterward, we conducted several continual learning experiments using the proposed deep learning architecture.

In the second experiment, six tasks (2 classes per task) incremental learning using the 12 classes MIT-BIH dataset is implemented. As illustrated in Table 3 and Figure 5 (a), the model achieved an overall accuracy (OA) of 99.92% at the first task, then reached the accuracy of 93.45% at the last classification task after performing all the tasks incrementally. A small decrease of 6.47% in accuracy happened during the continual learning process.

Table 3. Overall accuracy in [%] obtained for six tasks (2 classes/task) MIT-BIH 12 classes dataset.

Task	One Step Learning	Overall Acc.	Avg. Acc.	F1-Score
1	99.04	99.92	99.92	99.75
2		99.27	99.63	95.13
3		94.49	98.16	93.86
4		97.78	99.44	96.15
5		97.38	99.48	97.10
6		93.45	98.91	88.55

In the third experiment, the performance of the proposed continual learning model is tested using the eight classes INCART dataset. For a unique task with the entire dataset, the model performed with an accuracy of 98.9%. During the four tasks incremental learning (2 classes per task), the model executed the first task with an accuracy of 99.94% and terminated the continual learning process with an accuracy of 98.04% in the last classification task as shown in Table 4 and Figure 5 (b).A decrease of 1.9% in accuracy is noticed after performing all the tasks sequentially.

Table 4. Overall accuracy in [%] obtained for four tasks (2classes/task) from INCART dataset.

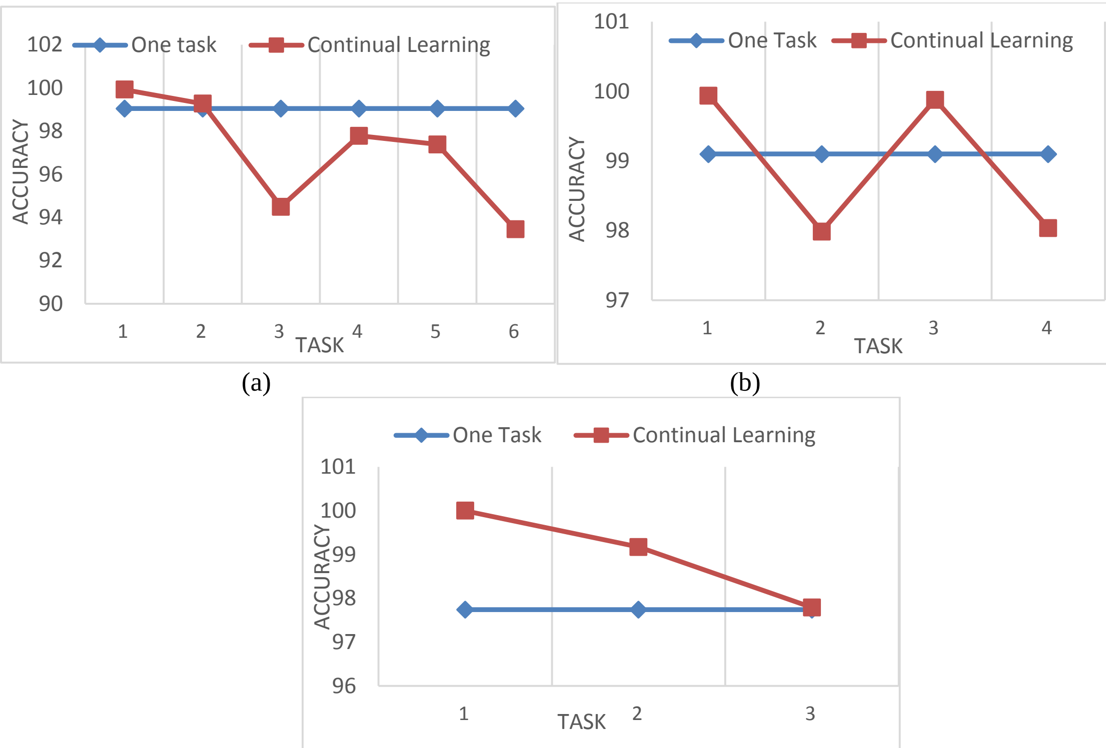
Task	One Step Learning	Overall Acc.	Avg. Acc.	F1-Score
1	98.9	99.94	99.94	99.26
2		97.99	98.99	93.32
3		99.88	99.96	82.82
4		98.04	99.51	86.96

The SVDB dataset is used to evaluate the proposed architecture in the fourth experiment. The model performed with an accuracy of97.74% on the entire dataset. During the incremental learning process, and to perform three tasks (2 classes per task), we divided the SVDB dataset into three sets. As illustrated in Table 5 and Figure 5 (c), the model achieved the first task with an accuracy of 100%, and reached the last task with an accuracy of 98.79% with a difference of 1.21%.

Table 5. Overall accuracy in [%] obtained for four tasks (2 classes/task) SVDB six classes dataset

Task	One Step Learning	Overall Acc.	Avg. Acc.	F1-Score
1	97.74	100	100	100
2		98.17	99.58	99.07
3		98.79	99.60	98.81

The fifth experiment is conducted to check the sensitivity of the proposed classification deep model to the number of classes per task. The twelve classes MIT-BIH dataset is divided into three sets, six tasks with 2 classes per task, four tasks with 3 classes per task, and three tasks with 4 classes per task. A continual learning experiment is executed on each set, and the results are collected and depicted in Figure 5. As shown in Figure 6, the number of classes per classification task does not affect the continual learning process in a significant manner.



(a) (b) (c)

Figure 5. Accuracy versus tasks for different datasets: (a) Accuracy for MIT-BIH database partitioned into 6 tasks with 2 classes by task, (b) Accuracy for ICART database partitioned into 4 tasks with 2 classes by task, (c) Accuracy for SVDB database partitioned into 3 tasks with 2 classes by task.

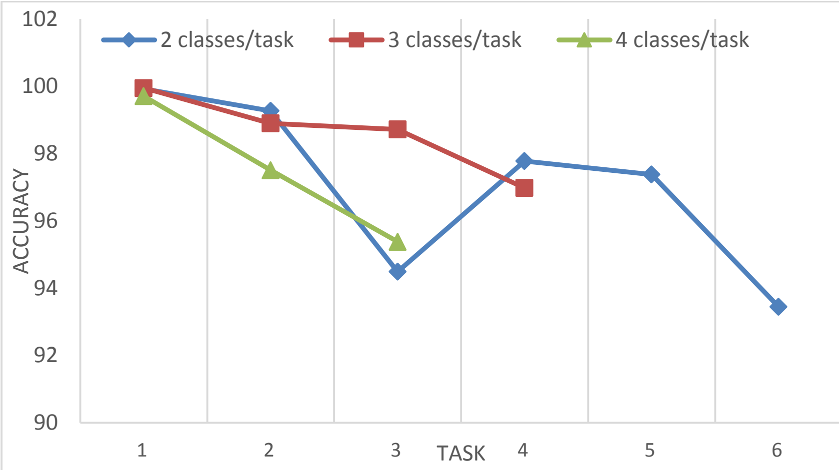


Figure 6. Sensitivity to the number of classes per task

The performance of the proposed architecture was also compared against the two state-of-the-art methods called elastic weights consolidation (EWC) [32] and Learning-without-forgetting approach for electrocardiogram heartbeat classification based on memory with task selector (LwF-ECG) [33]. Compared to the state-of-the-art methods EWC[32] and LwF-ECG [33], the Contrast-CLGAN method is significantly better for all datasets as shown in Table 6. The proposed modelmaintains its performance between the different classification tasks during the incremental learning process, demonstrating the strong capability of the Contrast-CLGAN method at preserving the acquired knowledge about the old classification tasks.

Table 6. Comparison with other methods.

Dataset		MIT-BIH			INCART			SVDB	
Method	Proposed	LwF [28]	EWC[27]	Proposed	LwF [28]	EWC[27]	Proposed	LwF [28]	EWC[27]
Task									
1	99.92	99.78	94.70	99.94	98.51	97.77	100	99.70	95.70
2	99.27	87.68	71.64	97.99	80.97	79.70	98.17	90.37	64.00
3	94.49	76.13	89.99	99.88	79.90	54.00	98.79	87.95	48.82
4	97.78	74.29	70.21	98.04	72.83	48.32	-	-	-
5	97.38	71.67	77.99	-	-	-	-		-
6	93.45	66.35	54.15	-	-	-	-	-	-

3.4 Discussion

From the obtained results, it is clear that the conversion of the 1D ECG signal into a 2D image and the insertion of a contrastive learning module at the top of the model have boosted the classification accuracy significantly. Memorizing knowledge about the old tasks is required to fight the forgetting phenomenon during the continual learning process. For this purpose, the GAN has the role of gathering and memorizing the necessary information about the old tasks in its weights. Once needed during the new task, the old tasks data is generated using the GAN model. Another utility of the GAN is to generate and augment the data of small classes, which is necessary for training a deep model.

IV. Conclusions and feature scope

Learning without forgetting remains a challenging task due to the difficulty in preserving the acquired knowledge of previous classification tasks when training the pre-trained model on a new classification task. In this work, a continual learning solution for ECG heartbeat classification is presented. The proposed deep learning model comprises three trainable modules. The first module learns to extract good features to distinguish between the different classes of the ECG heartbeat signal. The second module learns to increase the contrast between the different ECG heartbeat classes by using a contrastive learning strategy. The third module contributes to the continual learning process by preserving the previous dataset's knowledge by conserving their latent structures. Another contribution is the feature extraction sub-model which converts the 1D ECG heartbeat signal into a 2D image via a set of neural network layers to get good features from the EfficientNet CNN feature extraction layers. The experimental results obtained from the classification of different ECG signal datasets showed the efficiency of the proposed architecture to deal with learning without forgetting.

In this proposed research work, we presented a generative model based framework for training a deep model to learn a sequence of tasks without forgetting the acquired knowledge. In future work, we will propose new efficient techniques to keep the knowledge of the previously learned tasks and to boost the discrimination between the different classes in the feature space.

Funding
This research was funded by the Researchers Supporting Project, King Saud University, Riyadh, Saudi Arabia, grant number RSP-2021/ 69.

Conflict of interest
The author declare that they are no conflicts of interest.

References

[1]. A. Sahu, H. GM, M. K. Gourisaria, S. S. Rautaray, and M. Pandey, "Cardiovascular risk assessment using data mining inferencing and feature engineering techniques," International Journal of Information Technology, vol. 13, no. 5, pp. 2011–2023, 2021.
[2]. American Heart Association, "Heart disease, stroke and research statistics at-a-glance," Retrieved on April, vol. 1, p. 2016, 2016.
[3]. S. Mendis, P. Puska, B. Norrving, and World Health Organization, Global atlas on cardiovascular disease prevention and control. World Health Organization, 2011.
[4]. R. Mehra, "Global public health problem of sudden cardiac death," Journal of electrocardiology, vol. 40, no. 6, pp. S118–S122, 2007.
[5]. C. Ye, M. T. Coimbra, and B. V. Kumar, "Arrhythmia detection and classification using morphological and dynamic features of ECG signals," 2010, pp. 1918–1921.
[6]. C. Van Mieghem, M. Sabbe, and D. Knockaert, "The clinical value of the ECG in noncardiac conditions," Chest, vol. 125, no. 4, pp. 1561–1576, 2004.
[7]. Z. Peng and G. Wang, "A novel ECG eigenvalue detection algorithm based on wavelet transform," BioMed research international, vol. 2017, 2017.
[8]. X. Jiang, L. Zhang, Q. Zhao, and S. Albayrak, "ECG Arrhythmias Recognition System Based on Independent Component Analysis Feature Extraction," in TENCON 2006 - 2006 IEEE Region 10 Conference, Hong Kong, China, 2006, pp. 1–4. doi: 10.1109/TENCON.2006.343781.
[9]. S. Yang and H. Shen, "Heartbeat Classification using discrete wavelet transform and kernel principal component analysis," in IEEE 2013 Tencon - Spring, Sydney, Australia, Apr. 2013, pp. 34–38. doi: 10.1109/TENCONSpring.2013.6584412.
[10]. C.-C. Lin and C.-M. Yang, "Heartbeat Classification Using Normalized RR Intervals and Morphological Features," Mathematical Problems in Engineering, vol. 2014, pp. 1–11, 2014, doi: 10.1155/2014/712474.
[11]. P. Tadejko and W. Rakowski, "Mathematical morphology based ECG feature extraction for the purpose of heartbeat classification," 2007, pp. 322–327.
[12]. V. Mazaheri and H. Khodadadi, "Heart arrhythmia diagnosis based on the combination of morphological, frequency and nonlinear features of ECG signals and metaheuristic feature selection algorithm," Expert Systems with Applications, vol. 161, p. 113697, 2020.
[13]. Y. Bazi, M. M. Al Rahhal, H. AlHichri, N. Ammour, N. Alajlan, and M. Zuair, "Real-Time Mobile-Based Electrocardiogram System for Remote Monitoring of Patients with Cardiac Arrhythmias," Int. J. Patt. Recogn. Artif. Intell., p. 2058013, Nov. 2019, doi: 10.1142/S0218001420580136.
[14]. V. Mondéjar-Guerra, J. Novo, J. Rouco, M. G. Penedo, and M. Ortega, "Heartbeat classification fusing temporal and morphological information of ECGs via ensemble of classifiers," Biomedical Signal Processing and Control, vol. 47, pp. 41–48, 2019.
[15]. Y. Kaya, H. Pehlivan, and M. E. Tenekeci, "Effective ECG beat classification using higher order statistic features and genetic feature selection," 2017, Accessed: Aug. 25, 2020. [Online]. Available: https://www.biomedres.info/abstract/effective-ecg-beat-classification-using-higher-order-statistic-features-and-genetic-feature-selection-8385.html
[16]. L. B. Marinho, N. de MM Nascimento, J. W. M. Souza, M. V. Gurgel, P. P. Rebouças Filho, and V. H. C. de Albuquerque, "A novel electrocardiogram feature extraction approach for cardiac arrhythmia classification," Future Generation Computer Systems, vol. 97, pp. 564–577, 2019.
[17]. P. S. Kumar and S. Sivamani, "Numerical analysis and implementation of artificial neural network algorithm for nonlinear function," International Journal of Information Technology, vol. 13, no. 5, pp. 2059–2068, 2021.
[18]. A. R. Patil and S. Subbaraman, "Performance analysis of static hand gesture recognition approaches using artificial neural network, support vector machine and two stream based transfer learning approach," International Journal of Information Technology, pp. 1–12, 2021.
[19]. R. Karthik, D. Tyagi, A. Raut, S. Saxena, K. P. Bharath, and M. Rajesh Kumar, "Implementation of Neural Network and Feature Extraction to Classify ECG Signals," in Microelectronics, Electromagnetics and Telecommunications, vol. 521, G. Panda, S. C. Satapathy, B. Biswal, and R. Bansal, Eds. Singapore: Springer Singapore, 2019, pp. 317–326. doi: 10.1007/978-981-13-1906-8_33.
[20]. G. Chen, Z. Hong, Y. Guo, and C. Pang, "A cascaded classifier for multi-lead ECG based on feature fusion," Computer methods and programs in biomedicine, vol. 178, pp. 135–143, 2019.
[21]. S. B. Jadhav, V. R. Udupi, and S. B. Patil, "Identification of plant diseases using convolutional neural networks," International Journal of Information Technology, vol. 13, no. 6, pp. 2461–2470, 2021.
[22]. S. Agrawal, A. Chowdhary, S. Agarwala, V. Mayya, and S. Kamath S, "Content-based medical image retrieval system for lung diseases using deep CNNs," International Journal of Information Technology, pp. 1–9, 2022.
[23]. 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, doi: 10.1007/s13246-019-00815-9.
[24]. M. Jogin, M. Madhulika, G. Divya, R. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (CNN) and deep learning," 2018, pp. 2319–2323.
[25]. R. M. French, "Catastrophic forgetting in connectionist networks," Trends in Cognitive Sciences, vol. 3, no. 4, p. 8, 1999.
[26]. I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," arXiv preprint arXiv:1312.6211, 2013.

[27]. S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 6, pp. 1137–1149, 2016.
[28]. Z. Li and D. Hoiem, "Learning without Forgetting," IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 12, pp. 2935–2947, Dec. 2018, doi: 10.1109/TPAMI.2017.2773081.
[29]. N. Ammour, Y. Bazi, H. Alhichri, and N. Alajlan, "Continual Learning Approach for Remote Sensing Scene Classification," IEEE Geoscience and Remote Sensing Letters, 2020.
[30]. S. S. Sarwar, A. Ankit, and K. Roy, "Incremental Learning in Deep Convolutional Neural Networks Using Partial Network Sharing," IEEE Access, vol. 8, pp. 4615–4628, 2020, doi: 10.1109/ACCESS.2019.2963056.
[31]. J. Xu and Z. Zhu, "Reinforced Continual Learning," arXiv:1805.12369 [cs, stat], May 2018, Accessed: Jun. 18, 2020. [Online]. Available: http://arxiv.org/abs/1805.12369
[32]. J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," Proceedings of the national academy of sciences, vol. 114, no. 13, pp. 3521–3526, 2017.
[33]. N. Ammour, H. Alhichri, Y. Bazi, and N. Alajlan, "LwF-ECG: Learning-without-forgetting approach for electrocardiogram heartbeat classification based on memory with task selector," Computers in Biology and Medicine, vol. 137, p. 104807, 2021.
[34]. M. M. Al Rahhal, Y. Bazi, H. Almubarak, N. Alajlan, and M. Al Zuair, "Dense Convolutional Networks With Focal Loss and Image Generation for Electrocardiogram Classification," IEEE Access, vol. 7, pp. 182225–182237, 2019, doi: 10.1109/ACCESS.2019.2960116.
[35]. M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2019, pp. 6105–6114.
[36]. G. B. Moody and R. G. Mark, "MIT-BIH Arrhythmia Database." physionet.org, 1992. doi: 10.13026/C2F305.
[37]. V. Tihonenko, A. Khaustov, S. Ivanov, and A. Rivin, "St.-Petersburg Institute of Cardiological Technics 12-lead Arrhythmia Database." physionet.org, 2007. doi: 10.13026/C2V88N.
[38]. S. D. Greenwald, "The MIT-BIH Supraventricular Arrhythmia Database." physionet.org, 1992. doi: 10.13026/C2V30W.
[39]. J.-J. Wei, C.-J. Chang, N.-K. Chou, and G.-J. Jan, "ECG data compression using truncated singular value decomposition," IEEE Transactions on Information Technology in Biomedicine, vol. 5, no. 4, pp. 290–299, 2001.
[40]. "ECGPUWAVE v1.3.4." https://physionet.org/content/ecgpuwave/1.3.4/ (accessed Aug. 25, 2022).