Adaptive Partition of ECG Diagnosis Between Cloud and Wearable Sensor Net Using Open-Loop and Closed-Loop Switch Mode

ADARSHA BHATTARAI1, DONGMING PENG1, JASON PAYNE2, AND HAMID SHARIF1, (Fellow, IEEE)
1Department of Electrical and Computer Engineering, University of Nebraska–Lincoln, Lincoln, NE 68588, USA
2Department of Cardiovascular Medicine, University of Nebraska Medical Center, Omaha, NE 68198, USA
Corresponding author: Adarsha Bhattarai (abhattarai3@huskers.unl.edu)

ABSTRACT Remote electrocardiogram (ECG) diagnosis with continuous real-time or near-real-time performance via a wireless wearable computing system would have significant value since it will enable on-time alerts and interventions, leading to life-saving outcomes. In this paper, a novel integration of open-loop and closed-loop switch modes is proposed, where the entire ECG diagnosis process is accomplished in three steps. First, the R-peak detection algorithm for initial diagnosis in local devices is executed, and 100% of the transmission energy is saved when no abnormality is detected. Second, in the case of an abnormality being detected, the edge device performs two-dimensional convolution neural network (2D-CNN) classification on the ECG signals, leading to either open-loop or closed-loop mode transmission based on the seriousness of the ECG signals. Third, in the cloud server, the received ECG signals may be further analyzed with a more sophisticated classification algorithm. The ECG classification accuracy ranges from 92.7% to 99.1% depending on whether the analysis is executed locally (with reduced communication costs) or remotely (with increased computing resources). Overall, the ECG diagnosis process is partitioned into three components, including 1) irregular heartbeat detection, 2) 2-D CNN classification in the edge device, and 3) further classification in the cloud. The simulation results of such partition of ECG diagnosis in these layers supervised by the open-loop and closed-loop switch modes have demonstrated that the proposed system architecture can achieve efficient ECG diagnosis via wearable technologies with both reliable accuracies and reduced communication energy.

INDEX TERMS Electrocardiogram (ECG), two-dimensional convolution neural network (2D-CNN), open-loop and closed-loop switch modes.

I. INTRODUCTION Accurate and efficient remote diagnoses are the major research topics in the telemedicine industry [1]–[4]. These research efforts have been focused on artificial intelligence-enabled wearable sensors, management of physiological signals generated by the wearable sensors, and roles of wearable sensors in aiding real-time monitoring. Remote Electrocardiogram (ECG) diagnosis is one of those Internet of Things-based healthcare applications that consists of a wearable sensor and is a major research area.

A robust real-time or near real-time healthcare technology would provide rapid ECG analysis and on-time treatment to patients with diverse cardiovascular diseases. Consequently, large numbers of patients could be saved from critical situations, and this would increase life expectancy. Currently, state-of-the-art wireless wearable ECG technologies rely on Artificial Intelligence (AI) diagnostic algorithms which are developed based on software development [5]–[7]. Due to their computational complexities, they usually only work offline within a unidimensional architecture. In that architecture, ECG data is recorded and then analyzed with sophisticated, time-consuming software algorithms. Moreover, the health profile of every patient is unique, and their diagnosis requirements and the necessary procedures might vary from one another. Therefore, an ECG diagnosis system must adapt to the specific requirements and situations for diverse cardiovascular diseases. The hardware and software
components, as well as the wireless communication strategies in the wearable system, need to be carefully designed.

With the emergence of AI in computing servers with powerful processors, it is possible to classify cardiovascular diseases, such as arrhythmia, with up to 98.8% accuracy [6–[8]]. An intelligent way of integrating such AI techniques in the edge or local devices could play a vital role in adaptive ECG diagnosis making real-time communication with cloud servers non-mandatory [9]–[12]. If such devices could independently diagnose the ECG signals with reliable accuracy, significant transmission energy could be saved, and instant diagnosis would be possible with minimal delay. Since real-time and efficient ECG diagnosis is a high priority, we believe a smart communication architecture is necessary among processing layers that will contribute to minimizing delay in the diagnosis and saving the transmission energy. A local/edge device could perform the initial diagnosis of the patient locally without transmitting every bit of the ECG signal to the medical server. It could only transmit the preliminary results of diagnoses to the medical server saving both time and energy. If further sophisticated analysis at the server end is needed when a more severe or complicated situation occurs, a feedback mechanism could be developed. This mechanism fulfills the request by demanding more detailed ECG signal data. In this research, we attempt to develop a novel communication architecture that works on interactive feedback and commands between different processing layers. An open-loop communication scheme improves the system delay and conserves energy; and a closed-loop communication scheme facilitates feedback among layers to provide quick, in-depth diagnoses of ECG data.

We used an R-peak detection algorithm based on discrete wavelet transform for the initial ECG diagnosis in the wearable sensor [13]–[17]. Then we performed the classification of ECG signals using 2D-CNN in an edge device [8], [18]. The results obtained from the edge device were further analyzed by the cloud server if necessary. The accuracy of detection in the local device could be further improved by combining the peak-detection algorithm and embedded AI inference module. This combination provides efficient communication of data from the local devices, i.e., transmit only when the abnormality is detected. This would directly enhance the battery life of the local devices.

A computing platform hub is proposed, that deals with numerous ECG signals of heart patients residing in the same locality, such as retirement or assisted living facilities. An intermediate component between the cloud and the multiple edges and local devices may be necessary to aggregate and analyze such a high volume of data. Hub can provide effective healthcare and medical assistance in those retirement facilities and centers. Moreover, the hub is necessary to minimize the delay in real-time diagnosis if every patient uses the same gateway for data transmission. The hub could be a powerful edge device that deploys large-sized heterogeneous patient data to multiple data routes.

To the best of our knowledge, no existing work addresses the proposed combination of open-loop and closed-loop communication systems that governs the adaptive partition of ECG diagnosis in the five layers of the proposed paradigm.

II. LITERATURE REVIEW

The design of a robust E-health-care system requires a solid understanding of edge/fog, and cloud computing. Naha et al. [19], Mao et al. [20], and Charyev et al. [21] discussed the important role of edge/fog layer computing in reducing delay and energy consumption. Similarly, Shi et al. [22] defined the role of edge computing as the involvement of any processing platform between the data source and the cloud server that lies at the edge of the network. So, it can be any device that is capable of processing data before the data reach the cloud layer like a smartphone. Furthermore, Cisco Systems introduced the concept of fog computing, which they define as a horizontal, system-level architecture that distributes computing, storage, control, and networking functions closer to the users along a cloud-to-thing continuum [23]. Cisco UCS E-Series Servers are the fog servers manufactured by the company [24]. Moreover, they explain that the fog and cloud layers work together directly to enhance performance, whereas edge layers have an indirect relationship. Nonetheless, IBM views fog and edge computing as similar platforms [25]. Though there are differences of opinions regarding edge and fog computing, there is consensus that cloud servers are the strongest servers that lie above all computing layers but have higher latency and power consumption [21], [26]–[28].

To overcome the disadvantages of cloud layers, various fog-based systems have been demonstrated for e-healthcare monitoring [29]–[32]. Farahani et al. [31] used Raspberry Pi and Intel Edison as the fog computers involved in the data exchange with the cloud. These research works indicate that the fog layer could be a powerful processing platform that is dedicated to supporting cloud servers. It can be concluded that fog servers or fog computing platforms are not present to compete with but to aid the cloud servers in accuracy and performance.

Accurate ECG diagnosis has been a major research topic in the telemedicine sector. The manual feature extraction of ECG signals could be a very slow process in the presence of AI techniques. Previous studies have used various deep learning techniques to classify arrhythmia using ECG data obtained from the MIT-BIH Arrhythmia Database. Cheikhrouhou et al. [12], Yildirim et al. [33] and Ullah et al. [6] used 1D Convolutional Neural Network(1D-CNN). Similarly, Jun et al. [7] and Ullah et al. [34] used 2D Convolutional Neural Network(2D-CNN) and Singh et al. [35] used Recurrent Neural Networks (RNN). Likewise, Huang et al. [36] and Greco et al. [37] proposed a 2D-CNN classification of ECG signals using short-time Fourier transform (STFT). Rather than manually extracting the features of ECG signals, 2D spectrogram images are used to classify Arrhythmia types. Similarly, Mashrur et al. [8]
used 2D scalogram images to classify Normal and Atrial fibrillation heart conditions. Unlike Huang et al. [36] and Mashrur et al. [8], Ullah et al. [12] used 1D-CNN to classify Arrhythmia types and claimed a classification accuracy rate of 98.46%. These papers presented improved results on the classification of arrhythmia as compared to the peak detection algorithms. Yet the accuracy rate could be affected by the number of samples in training, the number of epochs, and other deep learning parameters. Although the authors presented their improved ECG signal classification with high accuracy, they haven’t discussed deploying their technique in real-world telemedicine applications. Importantly, it is necessary to answer the question of how these detection techniques could be integrated with the local, edge, and cloud computing architecture in real-time ECG diagnosis.

Gill et al. [9] presented a fog/edge and cloud architecture introducing the concept of pre-diagnosis in edge devices for heart patients. The performance parameters, average network usage time, average energy consumption, and average latency, were demonstrated in the presence and absence of fog/edge devices. The deployment of edge devices significantly improved system performance. Similarly, Moghadas et al. [10] put forward the concept of local, edge, and cloud computing. They used Arduino’s electronic board and an AD8232 sensor module as a local device. The local device retrieved the patient’s ECG signal and transferred the data to a Raspberry 3B+ used as an edge device. The data analysis was done using a K-Nearest Neighbor (KNN) algorithm with a high accuracy rate. Gill et al. [9] and Moghadas et al. [10] have contributed to the overall energy minimization and latency of the system. However, these works did not discuss power consumption in wearable sensor devices, which is an important consideration for real-time health monitoring applications.

Djelouat et al. [11] explained the importance of battery life in real-time ECG monitoring. They used a powerful multicore device as an edge device that performed most of the heavy tasks away from the cloud. Importantly, they addressed the current issues of the modern healthcare system. Nevertheless, this work mainly focused on the versatile edge device but did not explore the capabilities of local computing or the possible contribution of local devices in the local, edge, and cloud paradigm.

The energy spent in communication is much greater than the energy spent in local processing. Various work has been conducted to address this problem. Sahu et al. [2] illustrated the unique idea that intermittent transmission between local and edge computers increases the battery life. Likewise, Cheikhrouhou et al. [12] integrated an inference module of 1D-CNN in the edge layer enhancing the latency and power consumption. However, the possibility of embedding an inference module in a less powerful device, like wearable devices was not discussed. Similarly, Kim et al. [38] used an adaptive 12-bit Analog to Digital converter (ADC) to lower power consumption. The ADC scaled the sampling frequency value, which ultimately reduced the transmission energy.

Wearable devices have limited battery life and processing power. So, intermittent transmission, embedded machine learning, and adaptive sampling frequency could be a possible fit for a robust and efficient wearable device. Some research in open literature has individually discussed the various layers lying between the wearable sensor net and the medical server [10]–[12]. However, they have not addressed the design principles or architecture of the system that governs the data interaction and transmission schemes. We believe that a holistic scheme of signal processing and communications should be developed in such a way that they do not compromise the instant and accurate diagnosis. At the same time, provide optimal performance in terms of hardware resource efficiency and signal analysis reliability.

To sum up our reviews on those references mentioned above, we may draw several important points. First, the interaction of multiple layers could improve real-time diagnosis by preventing unnecessary data transmission and detecting symptoms before a life-threatening heart condition. Second, the other aspects include initiating a faster medical emergency response and providing feedback regarding the poor signal quality induced by the contact errors in sensors. Furthermore, we believe that the implementation of a real-world telemedicine application requires a more detailed explanation regarding the relationship and flow of diagnoses between those layers. Hence, our proposed system architecture emphasizes a detailed relationship between the components, which focuses on accurate ECG diagnosis with our peculiar idea of an open-loop and closed-loop switch mode.

### III. SYSTEM ARCHITECTURE

The proposed model of local, edge and cloud computing for adaptive ECG diagnosis consists of five major computing platforms. There are two other components between local, edge, and cloud computing called fog servers and hubs.

The graphical representation of the proposed system architecture is shown in Figure 1.

The processing power and resources available from local to cloud servers are in increasing order. Hence, an adaptive ECG diagnosis uses these resources based on the computational complexity of the problem at a given time. Further analysis of the components involved in this adaptive diagnosis is discussed below:

#### A. LOCAL DEVICE

A local device can be any electronic device that consists of ECG sensors and is continuously monitoring the ECG signals of the patient in real-time. In our model, the wearable sensor uses an embedded R-peak detection algorithm to measure the patient’s heartbeat every minute [13], [14], [16]. If abnormal heartbeats (i.e., arrhythmia) are detected, they are further analyzed by the locally present AI inference module. Though the AI classification carried out in local devices has a lower accuracy rate compared to edge and cloud servers, it enhances
the R-peak detection algorithm. In the case of no abnormality detected, the local device stays idle to save power.

B. EDGE DEVICE
In our local, edge, and cloud paradigm, edge device refers to any PDA (Personal Digital Assistant) device, like modern-day smartphones. Smartphones have considerably higher processing power than local devices. These devices receive the ECG signals sent by local devices. These one-dimensional ECG signals are converted into 2D-ECG signals by the continuous wavelet transform to form 2-D images called scalograms. These scalograms are classified using 2D-CNN [15], [17], [18]. This will classify whether the given frame of the ECG signals is a normal or abnormal sinus rhythm. In the case of an abnormality, edge devices directly contact the fog and cloud server for further investigation.

C. FOG SERVERS
These are the intermediate components present between the edge devices and the cloud. Any device that enhances communication with a cloud server can act as a fog server. Rahmani et al. [39] used a PandaBoard from Texas Instruments to enable the fog-based application to improve the latency and transmission. Similarly, Fog devices, such as Raspberry Pi 3 (Quad Core 1.2GHz CPU, 1GB RAM, and 802.11n wireless) and Netbook (Intel Atom processor 1.6GHz, 2GB RAM, and 802.11b/g/n wireless), have been used for fog assistance [27]. These servers are located at shorter distances providing higher computing resources than edge devices. The 2D-CNN classification could be carried out with a larger set of training data and other enhanced parameters for improved accuracy. Also, connecting to the cloud now and then could increase the power consumption and result in delays. So, edge devices should connect to fog servers and transmit the classification results and ECG frames for additional investigation. Fog servers could be used as an alternative to the cloud when the edge-cloud environment has poor communication [40].

D. CLOUD SERVERS
These servers are the destination of the transmitted ECG signals from local devices, edge devices, and fog servers for more sophisticated analysis if necessary. Amazon AWS, Google GCP, IBM Cloud, Oracle Cloud, and Rackspace are the major available cloud servers [21]. Cloud servers have the greatest computational power of all the system components. The data transmitted are further analyzed and classified using 2D-CNN with the highest accuracy rate. In any case, an additional set of data is required by the medical server, feedback is sent to the edge devices to retrieve the requested patient information. For example, a cloud server could request the specific ECG signal frame from the edge devices.

Based on characteristics, a fog server and a cloud server are compared in Table 1 [24], [41], [42]. Similarly, a local device and an edge device are compared in Table 2 [43].

E. ECG SENSORS IN LOCAL DEVICES
Some wearable ECG sensors present in the market are the SEEQ sensor by Medtronic, ZIO XT Patch by iRhythm Technologies and the wearable biosensor by Philips [44]. These sensors are FDA-approved ECG sensors and can be used to collect ECG signals. Details of one of the sensors Philip biosensor is given in Table 3 [45].
A. Bhattarai et al.: Adaptive Partition of ECG Diagnosis Between Cloud and Wearable Sensor Net

### F. HUB

Hub is another component in our proposed model that manages the volume of data coming from a larger number of patients residing in the same locality. With the limited bandwidth and communication constraints, it is necessary to manage the bulk of data coming from those patients. So, the hub lies between the cloud/fog servers and multiple edge and cloud devices. This acts as another computing platform dedicated to minimizing the overall delay of the system. The basic relationship between the five major components in the adaptive ECG diagnosis model is represented in Figure 2.

### G. OPEN-LOOP COMMUNICATION AND CLOSED-LOOP COMMUNICATION BETWEEN THE LAYERS

It is necessary to optimize the battery of the local device as much as possible by various means such as the periodic transmission of data and performing the initial detection locally. Here, open-loop communication is initiated when the layers are not continuously connected, and they communicate only when an abnormality is detected. However, in the case of severe arrhythmia detection, it is required to inform the medical team as soon as possible. Finally, the transmission of such data from local to edge to the cloud is expected to be continuous. The communication initiated in such conditions should be closed-loop communication where every layer in the system can request and send the data simultaneously.

### H. ADAPTIVE SAMPLING FREQUENCY IN THE LOCAL DEVICES

In an environment where the communication between the local and edge devices is poor, the high-quality transmission of ECG signals could increase the overall latency in the system. So, adaptive sampling of the ECG signals could be carried out to overcome such scarce communication bandwidth. The digital ECG signals obtained from the ADC in local devices could be further processed by the CPU to lower the sampling frequency [38].

### I. INFRASTRUCTURE DEVELOPMENT

The proposed ECG diagnosis needs the development of five major computing platforms. The cost of infrastructure development could be higher than the traditional remote healthcare system which relied on limited processing power and fewer computing platforms. Though the cost could be higher the proposed system model is not unrealistic. It could be implemented with present technology and provides a secure ubiquitous service for remote ECG monitoring, diagnosis, and fast response.

### IV. METHODOLOGICAL FRAMEWORK

Different components or layers in the local, edge, and cloud paradigm should be assigned tasks based on their
capacity, optimizing the results. For example, the battery life of the local device is limited, so the system should assign fewer loads to make the local device more power efficient. Moreover, in the adaptive ECG diagnosis, these layers should work together in a way that there are no significant delays, and the transmission is efficient. If an interactive collaboration among these layers exists, a better system could be developed. The system could contribute to minimizing system latencies, optimizing the energy efficiency of local devices, and accurate ECG diagnosis. More importantly, such a hardware-resource-efficient system will enable continuous real-time or near-real-time ECG analysis and diagnosis. Such diagnoses are consequential and will lead to early response and life-saving outcomes. We believe that developing the infrastructure for the interactive local device, edge/fog servers, and cloud servers is a worthy investment because it improves real-time telemedicine applications in many ways. Controlling the data traffic, detecting symptoms before serious health conditions, and predicting medical emergencies at the earliest possible moments can upgrade the remote diagnosis remarkably. Similarly, feedback path can resolve the ECG sensors contact error in remote ECG diagnoses. Based on these considerations, our work proposes various algorithms with the flow of tasks shown in Figure 3.

**A. DATASET EXPLANATION**

ECG signals used in the experiments were obtained from the MIT-BIH Arrhythmia Database and MIT-BIH Normal Sinus Rhythm Database respectively [46]–[48]. The online platform PHYSIOBANK ATM was used to download the ECG data sets. ECG signals of 5 patients suffering from arrhythmia were downloaded for a closed-loop supervision analysis. The time duration of those 5 different ECG signals was approximately 30 minutes and each signal consisted of 650,000 samples. It was downloaded in .mat format which was compatible with MATLAB.

Similarly, the dataset for the 2D-CNN evaluation consisted of ECG signals obtained from the MIT-BIH Arrhythmia Database and MIT-BIH Normal Sinus Rhythm Database. 30 individual records each from Arrhythmia Database and Normal Sinus Rhythm Database were used in the evaluation. Each ECG record had 65,536 samples with a sampling frequency of 360 Hz.

**B. USE OF R-PEAK DETECTION ALGORITHM**

For the initial detection of abnormality in the local device, we used the R-Peak detection algorithm that uses the undecimated discrete wavelet transform (DWT) [13]. Symlet 4 wavelet is used for the R-peak detection because it gives a better transformation to raw ECG signal and made R-peak detection easier [13]. This algorithm is a good fit for the local devices as per their processing capability.

The algorithm 1.1 demonstrates the first step of the flowchart represented by Figure 3.

After the abnormal ECG signals are detected using the R-peak detection algorithm in the local device, the specific frame of the ECG signals is transmitted to the edge device for further analysis. Before this happens, the local device should
Algorithm 1.1 Monitoring of ECG Signals in the Local Devices

Input: ECG signals $ecg(t)$ monitored by the wearable sensors

Output: Detection of abnormal heartbeats per minute and R-R time interval.

Step 1: Calculate the undecimated DWT of the $ecg(t)$ using symlet 4 wavelet.

Step 2: Obtain 4 levels of decomposition coefficients [a4, d1, d2, d3, d4].

Step 3: Reconstruct the ECG signal using inverse discrete wavelet transform (IDWT) using only {d3 and d4} detailed coefficients.

Step 4: Find the R-peaks of the QRS complex of given $ecg(t)$.

Step 5: Calculate heartbeats per minute after locating R peaks.

Step 6: If abnormal heartbeats per minute or R-R interval is detected proceed to algorithm 1.2.

As soon as the local device configures the severity of the detection, either of the communication modes is followed. Open-loop communication is preferred over closed-loop communication when an abnormality is very rare. The local device transfers the frame of ECG signals periodically. It transfers the detected frame of ECG signals to the edge device and stays idle to save power. If the local device detects a series of abnormalities, it starts the closed-loop communication where the local device informs the edge device and cloud server. Then a simultaneous communication is established between the layers to investigate the case further. Algorithms 1.3.1 and 1.3.2 explain the steps carried out in open-loop communication and closed-loop communication, respectively.

C. OPEN-LOOP AND CLOSED-LOOP TRANSMISSION

Let $\alpha$, $\beta$, $\mu$, and $\Phi$ be the four different zones representing four severity levels of the patient. Zone $\alpha$ represents the highest threat and zone $\Phi$ represents the lowest threat as follows:

- $\alpha$: Severe threat zone
- $\beta$: Attention zone
- $\mu$: Consultation zone
- $\Phi$: Safe zone

A consultation zone $\beta$ is not an emergency but the patient needs to be careful with their current lifestyle and a consultation with a cardiologist is recommended. Moreover, the severity index ($S_i$) is a function of $N_r$, $H_b$, $H_s$, $G_h$, and $P_s$ as follows:

$$S_i = f(N_r, H_b, H_s, G_h, P_s)$$

$N_r$: Number of abnormal R-peaks detected
$H_b$: Average heartbeats per minute
$H_s$: Score based on patient past diagnosis records
$G_h$: Score based on patient genetic history
$P_s$: Patient age

Based on the value of $S_i$ either open-loop or closed-loop is initiated. The remote diagnosis of a patient will fall into one of the zones, i.e.

$$\text{Switch Mode} = \begin{cases} \text{Closed} - \text{Loop}, & S_i \in \text{or} \beta \\ \text{Open} - \text{Loop}, & S_i \in \mu \text{or} \Phi \end{cases}$$

Zones $\alpha$ and $\beta$ would lead to the closed-loop transmission while zones $\mu$ and $\Phi$ would lead to open-loop transmission.

D. USE OF 2D-CNN IN EDGE DEVICE

When the edge device receives the frames of ECG signals through either open-loop communication or closed-loop communication, it will initiate the 2D-CNN classification [17] based on continuous wavelet transform. Algorithm
Algorithm 1.3.1 Open-Loop Transmission
Input: Frames of abnormal ECG signals that are diagnosed with low severity by the local device.
Output: One-way transmission to the edge device without getting feedback from the edge or cloud devices.

Step 1: Initiate open-loop transmission of abnormal ECG frames and inform the edge device.
Step 2: Transmit the abnormal data to the edge device.
Step 3: After the completion of the transmission stay idle to save the power.

Algorithm 1.3.2 Closed-Loop Transmission
Input: Frames of abnormal ECG signals that are diagnosed with high severity by the local device.
Output: Start simultaneous transmission of data between local and edge devices with feedback.

Step 1: Initiate closed-loop transmission of abnormal ECG frames and inform the edge device.
Step 2: Transmit the abnormal data to the edge device and wait for any command from the edge device.
Step 3: Edge device informs the cloud server and stays online between local and cloud and waits for any command from the cloud server.

Algorithm 1.4 Diagnosis in Edge Device
Input: Receive frames of ECG signals \( ecg(t) \) from the local device
Output: Classification of normal sinus rhythm and Arrhythmia

Step 1: Calculate CWT of the 1-D \( ecg(t) \)
Step 2: After the transformation, every frame of ECG signals is converted into a 2-D image (scalogram) using the absolute value of the CWT coefficients.
Step 3: Classify scalograms using 2D-CNN such as Alexnet for accurate ECG diagnosis.
Step 4: Report the classification result to the cloud/medical server.

Algorithm 1.5 Utilization of Fog Servers
Input: Frames of abnormal ECG signals that are pre-diagnosed by edge device and need further analysis.
Output: Classification of normal sinus rhythm (NSR) and Arrhythmia (ARR) with better accuracy rate than edge device.

Step 1: Use algorithm 1.4 with a higher number of training data in the 2D-CNN classification to improve diagnostic accuracy.
Step 2: Report the classification result to the cloud/medical server.

Let \( T_{em} \) be the time taken to transmit the ECG records of the patient from the edge device to the medical server. Similarly, \( T_{ef} \) be the time taken to transmit the same ECG records of the patient from the edge device to the fog server. The decision on the requirement of a fog server is based on the following condition:

\[
\text{Fog assistance} = \begin{cases} 
\text{Recommended,} & \text{Case 1} \\
\text{Not recommended,} & \text{Case 2} 
\end{cases} \tag{8}
\]

Case 1: \( T_{em} > T_{ef} \), Case 2: \( T_{em} < T_{ef} \).

F. EXPERT ANALYSIS AT CLOUD/MEDICAL SERVER
Hence, the final destination of ECG signals is the medical server where the decision is made regarding the patient’s heart condition. If the patient requires immediate attention, an emergency warning is sent to the medical team who would provide instant service to the patient, as shown in algorithm 1.6.

G. DETAIL PARAMETERS OF 2D-CNN
The applied Alexnet CNN [49] has a total of 8 layers. The detailed parameters of Alexnet CNN used in the ECG classification are shown in Table 4. This table gives more information on activation sizes and operation sizes. Convolutions, Rectified Linear Unit (ReLU), Max Pooling, fully connected layer, Dropout, and SoftMax are the various operations, functions, or layers present in Alexnet.
Algorithm 1.6 Further Processing in Cloud/Medical Server

**Input:** Frames of abnormal ECG signals that are pre-diagnosed by the edge device or fog servers and need further analysis.

**Output:** Accurate ECG diagnosis with the help of medical experts and results obtained from 2-D CNN classification.

**Step 1:** Use algorithm 1.4 with the higher number of training data in the 2D-CNN classification improving the accuracy of the diagnosis.

**Step 2:** An expert analysis based on their knowledge and the results obtained from the local, edge, and cloud platform.

**Step 3:** Provide medical service to the patient according to the condition of diagnosis.

H. FEEDBACK BENEFITS

Our novel idea of a closed-loop mode provides feedback facilities between the multiple layers enabling commands from one layer to another. We considered the following useful e-healthcare scenarios that could be implemented in real-life diagnosis:

1) ON-DEMAND SIGNAL QUALITY IMPROVEMENT

Transmission of ECG signals from local to the cloud includes multiple layers and occasionally encounter signal degradation with poor signal to noise ratio (SNR). This might result from contact errors or other factors. In such conditions, quick feedback can be transmitted from the medical server or any layer commanding the re-transmission of the signals. Alternatively, the command can be sent to a local device to increase the sampling frequency of the signals.

2) VERIFICATION OF EMERGENCY SITUATION

Edge devices have lower classification accuracy compared to fog or cloud servers. When an edge device detects a false life-threatening situation, such a state can be further investigated by the fog or medical servers. Hence, medical or fog servers can send feedback after the verification of a false alarm.

I. CONTRIBUTION OF SWITCH MODES IN THE NETWORK

1) TACKLING DELAY

The edge devices and local devices can filter out redundant transmissions. These transmissions include ECG records of healthy patients which are not required to be uploaded to the medical server. These records are detected by the local devices and edge devices using the R-peak detection algorithm and 2-D CNN inference model respectively. Therefore, these multiple processing platforms are present in the network model to reduce unnecessary transmissions and hence minimize the communication delay.

| No. | Name   | Type           | Description               | Activation size |
|-----|--------|----------------|----------------------------|-----------------|
| 1   | 'data' | Image Input    | 227x227x3 images          | 227x227x3       |
| 2   | 'conv1'| Convolution    | 96 11x11x3 convolutions   | 55x55x96        |
| 3   | 'relu1'| ReLU           |                            | 55x55x96        |
| 4   | 'norm1'| Cross Channel Normalization | with 5 channels per element | 55x55x96        |
| 5   | 'pool1'| Max Pooling    | 3x3 max pooling           | 27x27x96        |
| 6   | 'conv2'| Grouped Convolution | 2 groups of 128 5x5x48 convolutions | 27x27x256       |
| 7   | 'relu2'| ReLU           |                            | 27x27x256       |
| 8   | 'norm2'| Cross Channel Normalization | with 5 channels per element | 27x27x256       |
| 9   | 'pool2'| Max Pooling    | 3x3 max pooling           | 13x13x256       |
| 10  | 'conv3'| Convolution    | 384 3x3x256 convolutions  | 13x13x384       |
| 11  | 'relu3'| ReLU           |                            | 13x13x384       |
| 12  | 'conv4'| Grouped Convolution | 2 groups of 192 3x3x192 convolutions | 13x13x384       |
| 13  | 'relu4'| ReLU           |                            | 13x13x384       |
| 14  | 'conv5'| Grouped Convolution | 2 groups of 128 3x3x192 convolutions | 13x13x256       |
| 15  | 'relu5'| ReLU           |                            | 13x13x256       |
| 16  | 'pool5'| Max Pooling    | 3x3 max pooling           | 6x6x256         |
| 17  | 'fc6'  | Fully Connected | 4096 fully connected layer | 1x1x4096        |
| 18  | 'relu6'| ReLU           |                            | 1x1x4096        |
| 19  | 'drop6'| Dropout        | 50% dropout                | 1x1x4096        |
| 20  | 'fc7'  | Fully Connected | 4096 fully connected layer | 1x1x4096        |
| 21  | 'relu7'| ReLU           |                            | 1x1x4096        |
| 22  | 'drop7'| Dropout        | 50% dropout                | 1x1x4096        |
| 23  | 'fc8'  | Fully Connected | 2 fully connected layer   | 1x1x2           |
| 24  | 'prob' | Softmax        | softmax                    | 1x1x2           |
| 25  | 'output'| Classification Output | crossentropy with classes 'ARR' and 'NSR' | -               |

2) TACKLING HIGH TRAFFIC OF DATA AND DATA LOSS

The proposed five computing platforms work together in reducing the volume of communication data transferred from one node to another. The novel open-loop and closed-loop switch modes make that possible. It divides the data into various groups and prioritizes the transmission according to patients’ needs and conditions. The high-priority patient record is transmitted to the medical server through closed-loop mode without compromising the resources. In contrast, low priority data are processed by the local devices or edge devices and only a short report is transmitted to the medical server. Doing this not only reduces the bulk of data but also reduces the probability of data loss, data theft and data leaks.
V. SIMULATION RESULTS AND ANALYSIS

In our simulations, we used the ECG signals from the MIT-BIH Arrhythmia Database. In this section, the R-peak detection algorithm and 2D-CNN classification are tested and implemented separately using MATLAB R2020b. These simulation results demonstrate that the whole process of ECG diagnosis can be accomplished in the three components, including 1) irregular heartbeat detection in local devices and 2) 2-D CNN in edge devices, and 3) further classification in fog/cloud servers. The simulation results verify the feasibility of such partition of ECG diagnosis in these layers supervised by the open-loop and closed-loop switch modes. Results demonstrate that the proposed system architecture can achieve efficient ECG diagnosis via wearable technologies with both reliable accuracies and reduced communication energy. In the proposed system, ECG data is transmitted only if necessary, and analyzed in edge devices with AI inference module to achieve both accuracy and efficiency.

A. R-PEAK DETECTION

Three R-peak detection experiments were carried out using different ECG signals length of 10 seconds, 60 seconds, and 30 minutes. The R-Peak detection algorithm was proposed for the local device to calculate heartbeats per minute and R-R intervals. The algorithm used very limited computing resources and displayed abnormalities instantly. When the local device detected unusual heart rate values and R-R intervals, as shown in Figure 4, it initiated either open-loop communication or closed-loop communication with the edge device based on the patient’s degree of severity.

B. CLOSED-LOOP AND OPEN-LOOP INITIATION

Considering the 5 patients placed under the closed-loop communication model, their ECG signals were supervised continuously for 30 minutes, as shown in Figure 5. These patients belonged to either the severe threat zone or attention zone. If no abnormalities or rare abnormalities were present in these patients, open-loop communication would have been initiated. In this loop, there wouldn’t be any continuous transmission, and feedback saving the transmission energy.

C. 2D-CNN CLASSIFICATION

Further investigation of ECG signals in edge devices and servers was carried out using the 2D-CNN where the given ECG signals are diagnosed with high accuracy. Two different tests, each consisting of three different experiments were carried out. They were based on the number of training and testing images, and the number of samples taken to form the scalogram. 500 samples of ECG signals were used in the first 3 experiments to form each scalogram. 250 samples of ECG signals were used in the subsequent 3 experiments. This change led to an increase in classification accuracy. Therefore, in our proposed system architecture, various layers could adjust the scalogram sampling based on their processing power to improve the diagnosis accuracy.
These simulation results show several important points that support the feasibility of the partitioning algorithm as presented in the previous section of the paper. First, the AI training algorithm requires a higher amount of data (e.g., ECG scalogram images) to achieve reliable classification accuracy. Second, while the training process is completed, the AI inference algorithm may work with significantly fewer complexities while delivering accurate analysis performance. Third, when deployed in local embedded devices, the AI inference module can work with various simplified parameters, such as different sampling frequencies, to further achieve hardware resource efficiencies while still maintaining high accuracy results.

Several scalograms were used as training and testing images to distinguish between normal sinus rhythm and arrhythmia. Scalogram formed from 500 ECG samples is shown in Figure 7. Similarly, scalogram formed from 250 ECG samples is shown in Figure 6. Classification results were obtained using the Alexnet CNN [50]. The highest accuracy of 99.1% was obtained as shown in Figure 8. Likewise, Table 5 summarizes the classification accuracy of tests performed.

The experimental results suggest that the classification result improved after increasing the number of both training and testing images. According to the capacity of the devices or servers, these parameters could be changed to obtain optimum results. We conclude that experiments 1 and 4 are ideal for the initial ECG diagnosis in the local device. The inference module based on experiments 1 and 4 could be embedded in the local device to improve diagnosis accuracy.

On the other hand, as the processing power increases from edge devices to fog servers and cloud servers, a furthermore sophisticated machine learning algorithm could be used. They can have a greater number of training and testing images that could be used to enhance the more in-depth diagnosis. Such diagnosis often incurs much higher computing complexities and complicated floating-point calculations and may not be otherwise implementable in edge devices.

**D. ANALYSIS OF EXPERIMENTAL RESULTS**

1) LOCAL DEVICE COMPATIBILITY

The experimental results in section IV/A suggest that the R-Peak detection algorithm can be easily implemented in local devices by spending limited computing resources. Therefore, the algorithm best fits the initial ECG diagnosis and is proposed for the less powerful local device.

2) ACCURATE AND CONFIGURABLE CNN DIAGNOSIS

The use of 2D-CNN to diagnose the heart patient, prove to be very accurate and efficient. As suggested in Table 5 we

---

**TABLE 5. Results and analysis of 2D-CNN on ECG records from MIT-BIH database.**

| Experiment | Type     | Number of ECG records | Scalograms obtained from each record | Total Scalograms | ECG samples in each scalogram | Trained scalogram images | Tested scalogram images | Classification Accuracy | Prospective Device     |
|------------|----------|------------------------|--------------------------------------|-----------------|-------------------------------|-------------------------|--------------------------|------------------------|-------------------------|
| 1          | Arrhythmia | 30                     | 10                                   | 300             | 500                           | 250                     | 50                       | 92.7%                  | Local Device           |
|            | NSR      | 30                     | 10                                   | 300             | 500                           | 250                     | 50                       |                        |                         |
| 2          | Arrhythmia | 30                     | 50                                   | 1500            | 500                           | 1260                    | 240                      | 97.9%                  | Edge Device/ Hub       |
|            | NSR      | 30                     | 50                                   | 1500            | 500                           | 1260                    | 240                      |                        |                         |
| 3          | Arrhythmia | 30                     | 130                                  | 3900            | 500                           | 3276                    | 624                      | 98.9%                  | Medical Server/ Fog Server |
|            | NSR      | 30                     | 130                                  | 3900            | 500                           | 3276                    | 624                      |                        |                         |

**FIGURE 6. Scalogram with 250 ECG samples.**

**FIGURE 7. Scalogram with 500 ECG samples.**
can reconfigure the testing, and training images and samples in the scalogram according to the computing resources. It is concluded that for an embedded system like a local device and edge device it is better to use the inference module of 2D-CNN. Similarly, for powerful fog servers and medical servers, a large number of training data sets can be used to train the CNN model. Hence, powerful processing platforms can be used to achieve accurate diagnosis results.

3) REDUCED DATA TRAFFIC, DATA LOSS, AND DELAY
Open-loop switch mode in the local device and edge device reduced the volume of data in the network by excluding the healthy datasets. This resulted in a significant reduction in the overall traffic of the data. Moreover, the probability of data loss and network delays was reduced. In contrast, the closed-loop mode was initiated to give quality service to unhealthy patients as demonstrated in Figure 5.

4) REDUCTION IN TRANSMISSION ENERGY
The proposed switch modes contributed to the reduction of transmission energy. When the normal ECG signals were detected by the local device or edge device it was ignored in the transmission. Only severe patients’ signals were transmitted and monitored as shown in Figure 5.

E. ROBUSTNESS IN REAL-TIME APPLICATION
There has been a significant transformation in wearable ECG sensors from the classic Holter monitor to the smart ECG wearable sensors like ZIO XT Patch, and the apple watch [44], [45], [51], [52]. These smart devices represented ECG sensors in the proposed work. Similarly, the modern-day smartphone represented an edge device with powerful processing power. Those smartphones can support wireless connections with local sensors, fog servers, and medical servers at the same time. They are also capable of holding inference modules of machine learning algorithms for instant diagnoses. Amazon AWS, Google GCP, IBM Cloud, Oracle Cloud, and Rackspace are available cloud servers that represented medical servers [21]. Cisco UCS E-Series Servers represented fog servers [24]. Powerful personal computers represented Hub. In this way, the proposed system can use currently available technology to implement ECG diagnosis in various processing platforms. The present technology is already resilient, and the same infrastructure could be used for the proposed real-time healthcare system. Additionally, the simulation results of R-peak detection algorithms, 2D-CNN diagnoses, and open-loop/closed-loop supervision demonstrate the feasibility of the suggested method for deployment in real-time telemedicine application.

VI. CONCLUSION
We presented a novel open-loop and closed-loop communication switch mode for efficient real-time ECG signal analysis, transmission, and diagnosis. In the existing literature, state-of-the-art wireless wearable ECG technologies rely on AI diagnosis that is based on software development. These AI algorithms usually feature high computational complexities...
and can only work offline while ECG data is recorded, offloaded, and then analyzed. In this paper, we aimed to exploit the advantages of embedded computing powers and elements, and partition and coordinate the multilayered algorithms among wearables computing and cloud computing platforms.

The open-loop and closed-loop switch communication mode takes advantage of the diversity of unequal importance of ECG signals and transmits these ECG signals with different strategies. When ECG signals represent normal or healthy situations, the open-loop communication scheme will take effect; When ECG signals represent potentially severe health problems, the closed-loop communication scheme will be deployed. In this paper, the entire ECG diagnosis process is accomplished in three steps. The R-peak detection algorithm for initial diagnosis in local devices is executed, and ECG data will not be transmitted when no abnormality is detected. Furthermore, if the abnormality is detected, the inference module of 2D-CNN classification is carried out in the edge device, leading to either open-loop or closed-loop mode transmission based on the seriousness of the ECG signals. Lastly, the received ECG signals could be further analyzed in the cloud server with a more sophisticated classification algorithm. Hence, this work presented the idea of ECG diagnosis partition in local, edge, and cloud computing with a series of corresponding algorithms.

Open-loop communication was preferred when the power optimization for the local device was a major concern, and the abnormalities were minor. Closed-loop communication was initiated when the patient’s arrhythmia was critical and required extra care. Interactive continuous monitoring was performed using feedback among different layers in this mode.

REFERENCES

[1] R. Barnes and K. Zvarikova, “Artificial intelligence-enabled wearable medical devices, clinical and diagnostic decision support systems, and Internet of Things-based healthcare applications in COVID-19 prevention, screening, and treatment,” Amer. J. Med. Res., vol. 8, no. 2, pp. 9–22, 2021.
[2] N. Sabu, D. Peng, and H. Sharif. “Dagnosis-Steganography-communication: An innovative integrated paradigm for ECG healthcare,” Social Netw. Comput. Sci., vol. 2, no. 04, pp. 1–22, Jul. 2021, doi: 10.1017/s2979-021-00072-6.
[3] D. Hurley and G. H. Popescu, “Medical big data and wearable Internet of Things healthcare systems in remotely monitoring and caring for confirmed or suspected COVID-19 patients,” Amer. J. Med. Res., vol. 8, no. 2, pp. 78–90, 2021.
[4] D. Turner and A. Pera, “Wearable internet of medical things sensor devices, big healthcare data, and artificial intelligence-based diagnostic algorithms in real-time COVID-19 detection and monitoring systems,” Amer. J. Med. Res., vol. 8, no. 2, pp. 132–145, 2021.
[5] J.-S. Huang, B.-Q. Chen, N.-Y. Zeng, X.-C. Cao, and Y. Li, “Accurate classification of ECG arrhythmia using MOMWT enhanced fast compression deep learning networks,” J. Ambient Intell. Humanized Comput., vol. 2, pp. 1–18, May 2020, doi: 10.1007/s12652-020-02110-y.
[6] A. Ullah, S. U. Rehman, S. Tu, R. M. Mehmood, Fawad, and M. Ehatisham-Ul-Haq, “A hybrid deep CNN model for abnormal arrhythmia detection based on cardiac ECG signal,” Sensors, vol. 21, no. 3, p. 951, Feb. 2021, doi: 10.3390/s21030951.
[7] T. Joon Jun, H. Minh Nguyen, D. Kang, D. Kim, D. Kim, and Y.-H. Kim, “ECG arrhythmia classification using a 2-D convolutional neural network,” 2018, arXiv:1804.00812.
[8] F. R. Mashrur, A. Dutta Roy, and D. K. Saha, “Automatic identification of arrhythmia from ECG using AlexNet convolutional neural network,” in Proc. 4th Int. Conf. Electr. Inf. Commun. Technol. (EICT), Khulna, Bangladesh, Dec. 2019, pp. 1–5, doi: 10.1109/eict48899.2019.9068806.
[9] S. S. Gill, R. C. Arya, G. S. Wander, and R. Buyya, “Fog-based smart healthcare as a big data and cloud service for heart patients using IoT,” in Proc. Int. Conf. Intell. Data Commun. Technol. Internet Things (ICICT), 2019, pp. 1376–1383.
[10] E. Moghadash, J. Rezazadeh, and R. Farahbakhsh, “An IoT patient monitoring based on fog computing and data mining: Cardiac arrhythmia usecase,” Internet Things, vol. 11, Sep. 2020, Art. no. 100251.
[11] H. Djelout, M. Al Disi, I. Boukhenoufa, A. Amira, F. Bensaali, C. Kotronis, E. Politä, M. Nikolaidou, and G. Dimitrakopoulos, “Real-time ECG monitoring using compressive sensing on a heterogeneous multicore edge-device,” Microprocessors Microsystems, vol. 72, Feb. 2020, Art. no. 102839.
[12] O. Cheikhrouhou, R. Mahmud, R. Zouari, M. Ibrahim, A. Zaguida, and T. N. Gia, “One-dimensional CNN approach for ECG arrhythmia analysis in fog-cloud environments,” IEEE Access, vol. 9, pp. 103513–103523, 2021.
[13] J. Park, J.-S. Park, S.-W. Lee, and U. Park, “R peak detection method using wavelet transform and modified Shannon energy envelope,” J. Healthcare Eng., vol. 2017, pp. 1–14, Jul. 2017, doi: 10.1155/2017/4901017.
[14] V. Alarcon-Aquino and J. Barria, “Change detection in time series using the maximal overlap discrete wavelet transform,” Latin Amer. Appl. Res., vol. 39, no. 2, pp. 145–152, 2009.
[15] L. Zhu, Y. Wang, and Q. Fan, “MODWT-ARMA model for time series prediction,” Appl. Math. Model., vol. 38, nos. 5–6, pp. 1859–1865, Mar. 2014, doi: 10.1016/j.apm.2013.10.002.
[16] A. Bhattarai et al.: Adaptive Partition of ECG Diagnosis Between Cloud and Wearable Sensor Net
[17] X.-Y. Zhang and R.-J. Zhang, “The technology research in decomposition and reconstruction of image based on two-dimensional wavelet transform,” in Proc. 9th Int. Conf. Fuzzy Syst. Knowl. Discovery, Chongqing, China, May 2012, pp. 1998–2000, doi: 10.1109/fskd.2012.6234204.
[18] R. K. Naha, S. Garg, D. Georgakopoulos, P. P. Jayaraman, L. Gao, X. Xiang, and R. Ranjan, “Fog computing: Survey of trends, architectures, requiremements, and research directions,” IEEE Access, vol. 6, pp. 47980–48009, 2018.
[19] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing: The communication perspective,” IEEE Commun. Surveys Tuts., vol. 19, no. 4, pp. 2322–2358, 4th Quart, 2017.
[20] B. Charyyev, E. Arslan, and M. H. Gunes, “Latency comparison of cloud datacenters and edge servers,” in Proc. IEEE Global Commun. Conf., IEEE, Dec. 2020, pp. 1–6, doi: 10.1109/ gLOBECOM42020.2020.9322406.
[21] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: Vision and challenges,” IEEE Internet Things J., vol. 3, pp. 637–646, May 2016.
[22] (2017). OpenFog Reference Architecture for Fog Computing Produced by the OpenFog Consortium Architecture Working Group. Accessed: Mar. 21, 2022. [Online]. Available: https://site.ieee.org/denver-computing.com/c/en/us/products/ servers-unified-computing/ecs-e-series-servers/index.html
[23] A. Banafa. (Aug. 25, 2014). What is Fog Computing? Cloud Computing News. Accessed: Mar. 21, 2022. [Online]. Available: https://www.cloudcomputingnow.com/fog-computing/2014/08/fog-computing/
[24] V. Kumar, A. A. Laghari, S. Karim, M. Shakir, and A. A. Brohi, “Comparison of fog computing & cloud computing,” Int. J. Math. Sci. Comput., vol. 5, no. 1, pp. 31–41, Jan. 2019.
[25] G. L. Santos, P. T. Endo, M. F. Ferreira da Silva Lisboa Tigre, L. G. Ferreira da Silva, D. Sadow, J. Kelner, and T. Lynn, “Analyzing the availability and performance of an e-health system integrated with edge, fog and cloud infrastructures,” J. Cloud Comput., vol. 7, no. 1, pp. 1–22, Dec. 2018, doi: 10.1186/s13677-018-0118-3.
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 2001.

A. Rashkovska, M. Depolli, I. Tomašić, V. Avbelj, and R. Trobec, “A comparative study of Arduino, raspberry pi and ESP8266 as IoT development board,” Int. J. Adv. Res. Comput. Sci., vol. 8, no. 5, pp. 2350–2352, 2017. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=ED553113&asrn=ED553113&ranSiteId=5ea566852f55c76f7fa91b2f563c02c6&ti=aripin21
df=hi&nd=y&tenv=prod/eh&sid=62489c7f7e6e6a0778f047e17d0c6732&vid=0&contextAPI=fb24c63f8f4031f66c8f6d76717f6c05&ctx_ver=Z39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-8&ctx lens=info%3Aofi%2Ffmt%3Akev%3Amtx%3Ajournal%3ArtICLE&idagent%3Asimr000102&local_base%3Alan2010&spf=pfr

A. Bhattarai received the B.S. degree in electrical and electronics engineering from Istanbul University, Turkey, in July 2021. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Nebraska–Lincoln. His primary research interests include digital signal processing, the Internet of Things, and security in wireless sensor networks.

Dongming Peng received the Ph.D. degree in computer engineering from Texas A&M University. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Nebraska–Lincoln. He has published over 100 technical articles in the research areas of digital image processing, multimedia security, wireless sensor networks, cybersecurity, computer architecture, and related topics. He has conducted various research projects in these areas with special contributions in energy-efficient wireless and secure communications in multimedia sensor networks.

Hamid Sharif (Fellow, IEEE) is currently a Charles J. Vranek Professor with the Department of Electrical and Computer Engineering, University of Nebraska–Lincoln (UNL), Lincoln, NE, USA. He is also the Director of the Advanced Telecommunication Engineering Laboratory (TEL), UNL. He has authored or coauthored over 400 research papers in national and international journals and conferences and has been serving on many IEEE and other international journal’s editorial boards. He has received a number of research awards and best articles. He is a Distinguished Lecturer for the IEEE Vehicular Technology Society.

ASHOK SHARMA

[ADARSHA BHATTARAI received the B.S. degree in electrical and electronics engineering from Istanbul University, Turkey, in July 2021. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Nebraska–Lincoln. He has published over 100 technical articles in the research areas of digital image processing, multimedia security, wireless sensor networks, cybersecurity, computer architecture, and related topics. He has conducted various research projects in these areas with special contributions in energy-efficient wireless and secure communications in multimedia sensor networks.

JASON PAYNE received the degree (Hons.) in 2004. He is currently an Assistant Professor with the Department of Internal Medicine, University of Nebraska Medical Center. Moreover, he is a Cardiovascular Electrophysiology Specialist and has more than 18 years of diverse experiences, especially in cardiac electrophysiology and cardiovascular disease (cardiology).]