International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys.14, No.1 (Jul-23)

http://dx.doi.org/10.12785/ijcds/140114

Simulation of Multimedia Data Transmission Over WSN Based on MATLAB/SIMULINK

Akram Abdulmawjood Dawood[1], Azhar Sabah Abdulaziz[2], Amar Idrees Daood[1] and Qutaiba Ibrahim Ali[1]

[1]*Computer Engineering Department,College of Engineering, University of Mosul, Mosul, Iraq*
[2]*Software Department,College of Information Technology, Ninevah University, Mosul, Iraq*

Received 9 Jul. 2022, Revised 6 May. 2023, Accepted 8 May. 2023, Published 1 Jul. 2023

Abstract: The wireless sensor network (WSN) design requires investigating the effect of the transmit/receive operations on data quality. Having a convenient simulation model that facilitates and speeds up the work is a challenging task. The MATLAB/SIMULINK is considered as a handy tool that mimics real events for many applications. Unfortunately, the complete WSN model for multidimensional signals is not provided. Therefore, this paper introduces the MATLAB/SIMULINK model to deal with 1-D and 2-D signals in WSN. Many simulation blocks and codes have been programmed in this work to facilitate multimedia transmission over the WSN. The proposed multimedia WSN (MWSN) model will help WSN designers to plan and test their topology, taking into account most parameters and effects before implementing the network in real life. Hence, the WSN design time and cost will be reduced, as well as the malfunctioning will be avoided. Noise ,interference and path loss were the major parameters that affect the quality of the transferred data. Those parameters are experimented in this work using different multimedia signals and the received signal quality is measured using peak signal-to-noise ratio (PSNR), and bit error rate (BER). Additionally, different channels that add many levels of interference and signal-to-noise ratio are used to imitate reality. The ultimate goal of this research is to model the physical layer and modify the internal architecture of WSN nodes to deal with multimedia signals. Therefore, floating-point re-quantization and buffers are invoked to cope with the design requirements. The resultant simulation model is designed to be scalable and easy to expand for many different WSN topologies. The proposed MWSN has visualization features, in addition to the assessment tools that simplify results' analysis. Besides, the availability of tuning its parameters can help researchers to adopt it for different channels.

Keywords: Wireless Sensor Network (WSN), multimedia WSN, MATLAB/ SIMULINK,Signal-to-Noise Ratio (SNR), Digital Communication, Bit-Error Rate (BER).

1. Introduction

Lately, wireless sensor networks(WSN) have prevailed in many applications due to their practicality, minimized power consumption and low cost. They have the potential to be the dominant tool in many disciplines such as military, monitoring surveillance, healthcare, crisis and disasters management, dangerous environments detection, industrial world, control systems, tracking applications, traffics balance, and road hazard exploration. Due to the recent improvements in technology, WSN are designed to capture different types of signals, using sensors that can measure not only one dimensional signals but also two dimensional signals. For example in a natural crisis, WSN could be distributed flexibly using movable drones to detect human beings by a thermal-heat camera(2-D signal) and call back the rescue teams. After that, the WSN could be applied to measure temperature, blood pressure and even ECG or EEG(1-D signals) of the rescued people. The measured signals are transmitted to the specialist to perform a physical assessment and evaluation [1][2]. Over the last decade, researchers have spent a lot of effort to improve the WSN design. However, the implementation of WSN could be a very problematic issue due to the difficulty of meeting the design requirements of each application. A few papers have proposed to simulate the WSN in different techniques and platforms. The eventual intention of the simulation is to make sure that the designed nodes of WSN meet the requirements. Additionally, the simulation can measure the performance and assess the reliability of the designed system. The simulation of the WSN is still an open field and needs more work and improvement. Therefore, this paper proposes a WSN simulation using MATLAB/SIMULINK.

This paper discusses a WSN simulation using MATLAB/Simulink for red different types of signal, 1-D and 2-D signals. Several types of signals were transmitted and exposed to noise with different values using Bluetooth technology. The paper discusses the detailed implementation

E-mail address: akram.dawood@uomosul.edu.iq,azhar.abdulaziz@uoninevah.edu.iq, amar.daood@uomosul.edu.iq,qutaibaali@uomosul.edu.iq

of the Matlab simulation for 1-D(signal or voice) and 2-D signals(image). The designed simulation considers the effect of noise and interference on signal and image quality.The proposed simulation using MATLAB/Simulink blocks and functions are described in detail.

The paper is organized in eight sections. Section 2 discusses the previous simulation approaches. In section 3, the proposed general structure of the multimedia WSN (MWSN) simulation is introduced. Section 4 discusses the transmitter simulation steps. The channel simulation approach is considered in section 5. In section 6, the proposed MWSN receiver is explained. The experimental results are listed and discussed in section 7 and the final conclusion is placed in section 8.

2. Literature Review

In this section, some of the existing literature that are related to WSN simulation are discussed. Different software environments have been used to mimic different circumstances and various situations for WSN, which may be carried out by a complete modeling, a test-bed or a mathematical analysis. There are two types of simulation: general and specific. The first type of simulator is a general simulator, which is designed to simulate different functions and tasks. The environment of this simulator is designed to support different WSN functions. On the other hand, the second type of simulators is designed specifically to imitate the WSN environment [3],[4].

The Network Simulators, NS-2 and NS-3, which are object-oriented programming written entirely in C++ language, are widely used in the WSN field. The amount of contribution code for this simulator, which supported different types of network protocols, is huge. More specifically, there are some packages developed to support WSN functions such as Mannasim Framework, NS2-MIUN, and SensorSim [5], [6].

The authors in [7] constructed multiple sensing applications using the NS3 simulator. They have tested two types of data monitoring: periodic data and event-based data monitoring. They have carried out the simulation with different numbers of nodes with several arbitrary communications techniques. The researchers in [8] studied the performance of three routing protocols, namely, Ad Hoc On-demand Distance Vector, Dynamic Source Distance Vector, and Optimized Link State Routing. These routing protocols were tested using an NS3 simulator. The packet delivery ratio, data throughput, packet loss, and the delay of the transmission were examined and analyzed under the NS3. The researchers in [9] used NS-2 modeling software to study the WSN protocol throughput. They studied a specific case of wild animals' health tracking system, namely tiger vital signs. Eight mobile nodes are used to gather behavioral data, and send them to a server. They claimed that their proposed model was efficient regarding packet delay ratio (PDR) and energy. Routing protocol that relies on cross-layer operations was investigated in [10] using the NS-2 simulator. A stable routing was introduced by tackling the collision problem by changing the routing decision rule in an ad-hoc paradigm. The performance of the proposed approach was shown to be improved as compared to other routing approaches.

The proposed work in [11], was based on the TOSSIM tool to solve the time delay and power consumption through the WSN nodes' clustering. A fuzzy-neural hybrid model was introduced to improve the network throughput, which led to an increase in the WSN lifetime and decreased message delay. In [12], neural network had been implemented to study the packet end-to-end cost using TOSSIM. Researchers showed that distributed hardware over motes of WSN can be used to reduce expenses, such as time delay and memory consumption, as well as increase security factors.

Lakarani *et al.* [13] used Netsim to compare the routing energy consumption for different protocol types. They discovered that the AODV consumes less power for the same setting as compared to RPL. Meanwhile, the study in [14] compared the delay and energy consumption between AODV and RPL using Netsim for lossy WSN networks. They discovered that the AODV consumes 2.643% less energy than the RPL, and the latter protocol had more average delay time based on their conducted experiments.

OPNET is a commercial software that is used to simulate different types of wireless networks functions. It is developed and presented by OPNET Technologies, Inc. OPNET modeler uses a discrete event engine with a parallel simulation kernel, which supports two different operating systems: Windows and Linux. The OPNET simulator offers an object-oriented programming approach and hierarchical modeling. Additionally, it provides the facility of customized functions where it can be developed from scratch [15], [16], [17], [18]. For example, the researchers in [19] used OPNET and ZigBee to implement WSN to perform water quality monitoring. The simulation was carried out to track pH level and Turbidity. OPNET modeler was used in [20] to simulate the interaction of a multi-hop wireless sensor network using Markov random field (MRF) mathematical analysis. Uniform communication was adopted to imitate the traffic of the WSN, measure the validity of MRF and perform traffic inference. The researchers in [21] designed a validation model of WSN using OPNET modeler to determine bad nodes in distributed WSN with increasing numbers of the nodes in the designed model. The proposed simulation was implemented with 50 nodes using ZigBee WSN. The author in [22] developed a simulation model of energy consumption for WSN nodes in OPNET. A mathematical model was proposed to present the energy model based on the RM battery life of the WSN node.

The OMNeT++ is considered as a framework for network simulations purposes. It has a generic architecture with flexible tools that facilitate introducing new simulation scenarios. It has a very portable framework, hence it is compatible with different operating systems such as MAC , Linux, and Windows OS [23], [24], [25], [26]. The researchers in [27] examined five MAC layer protocols. They have created two scenarios using OMNET++ simulator with different numbers of nodes and power consumption. Power consumption, and packet congestion were computed to perform the analysis study. In [28] temperature and humidity have been attached to the Omnet++ simulator where the impact of these factors have been analyzed on Received Signal Strength Indicator.

MATLAB/SIMULINK can be considered as a general simulink because it has a wide spectrum of simulation tools and different functions. It can be programmed in two modes to design parallel or serial events. MATLAB architecture has rich tools to simulate both wired and wireless network functions [29], [30], [31], [3]. The authors in [32] studied the performance of ZigBee transceiver in Matlab Simulink by designing home area networks to connect smart meters and home appliances. Additive white Gaussian noise was selected to simulate the channel environment noise and the bit error rate with signal to noise ratio was analyzed to test the performance. In [33] routing strategy and path planning were examined using Matlab/Simulink. The main focus was on the path planning of multiple mobile nodes where K-mean clustering technique was used to cluster the node and optimize the best bath of each node. The Artificial Bee Colony (ABC) and Grey Wolf Optimizer (GWO) algorithms were implemented to cluster the WSN nodes more efficiently in [34]. The main concern in this work was the end-to-end delay time and WSN energy. The researchers used Matlab/Simulink to simulate the packet delivery ratio (PDR), then compared it to the famous LEACH protocol.The EEG signals, which are captured using the WSN sensors, are fed the Convolutional Neural Network (CNN) in the base station to detect arrhythmia, which was proved to be more accurate than the traditional ANN.

Table I summarizes the main aspects of the state-of-the-art literature that exploit the WSN simulation tools, showing the contribution of the proposed model in this regarding those attributes. The used simulation environment, and the OSI layer that each approach dealt with are listed. Besides, the table shows whether a case study was presented or not. Flexibility of the simulation states that the simulation can be expanded or its parameters can be altered for different applications. The signal dimensionalaty and the study type are included.

In spite of the necessity of considering the multi-dimensional signals in the IoT and WSN applications, it is noticed that a little effort has been accomplished to deal with 2-D signals. Therefore, building a simulation model that is able to cope with the multimedia signals was the motivation for this study. However, there are many challenges to adopt the 2-D signal, which start from the physical layer of the networking model. In this paper, the accurate

TABLE I. Related Work Attributes Summary

Ref. No.	Environment	Tunable Parameters	OSI Layer	Case Study	Signal Type	Study Type
[19],[20],[21]	OPNET	Yes	Transport	Yes	1-D	Routing Protocol
[22]	OPNET	Yes	Transport	Yes	1-D	Energy Consumption
[7]	NS3	Yes	Application	Yes	1-D	Scheduling
[8]	NS3	Yes	Transport	Yes	1-D	Routing Protocol
[27] , [28]	OMNET++	Yes	Physical	Yes	1-D	Power Consumption
[32]	MATLAB SIMULINK	Yes	Physical	Yes	1-D	Channel Noise
[33]	MATLAB SIMULINK	Yes	Transport	Yes	1-D	Routing Protocol
[9]	NS-2	Yes	Transport	Yes	1-D	Network Throughput
[10]	NS-2	Yes	Data Link	Yes	1-D	Collision avoidance and Energy Optimization
[11]	TOSSIM	Yes	Network	Yes	1-D	Power Consumption
[12]	TOSSIM	Yes	Transport	Yes	1-D	Security
[34]	MATLAB SIMULINK	Yes	Transport	Yes	1-D	Clustering
[13]	Netsim	No	Network	Yes	1-D	Routing Energy
[14]	Netsim	Yes	Network	Yes	1-D	Routing Throughput
Proposed Model	MATLAB SIMULINK	Yes	Physical	Yes	1-D & 2-D	Noise, Interference and Path loss

details of WSN node is described and implemented in Matlab/Simulink environment. This work gives more inside view into the actions played by the different sections of the node, which is considered as an additional contribution over the previous works.

3. The Multimedia WSN (MWSN) Simulation Structure

An overview of the simulation scenario that represents the proposed multimedia wireless sensor network (MWSN) simulation model is shown in Fig.1. A transmitter is followed by a wireless channel, which may add AWGN and/or a WLAN 802.11b interference signal. Then, the channel connects the transmitter and the receiver at the other end.

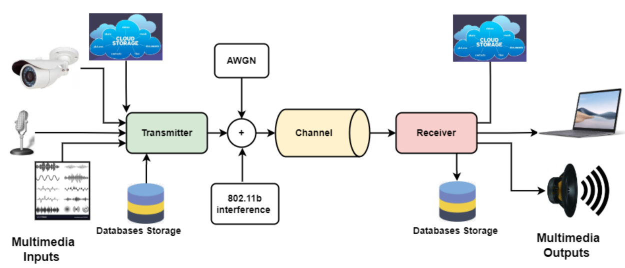


Figure 1. Block diagram of the general simulation scenario.

The proposed simulation aims to use the WSN to transmit and receive multimedia signals, which are expected to be 1-D signal and voice or 2-D image signals. The main concern is to simulate a multimedia wireless sensor network (MWSN) to measure the quality of the received signals after exposing it to noise and/or interference. It is an attempt to approach reality by introducing a MWSN simulation and help specialists to use such a handy tool. For 1-D signals, voice and speech, the information bits are encoded, modulated and sent to the channel directly.

On the other hand, 2-D signals, like images, need extra operations before transmission. In such a case, the 2-D signal is reshaped, re-quanized and buffered so it can be delivered to the transmitter. The proposed simulation can transmit a stored multimedia file using the already existing block in MATLAB simulink, in addition to a real time multimedia signal.

The MATLAB blocks used in the proposed simulation are depicted in Fig 2. The transmitter, the channel and the receiver are the main components of the design structure as shown in the “MATLAB Simulation Program” subsystem in the middle-top of Fig. 2. The inner components of each block are depicted in separate parts of Fig 2. The functions of those blocks are explained in the following sections.

4. The MWSN Transmitter Simulation

The transmitter can acquire any type of 1-D or 2-D signal, including voice, audio and image signals. When the audio signal happens to have 2-channels, the average of both channels is considered. The audio sensor can be a microphone or a stored audio file as far as the MATLAB/SIMULNIK regards. Regarding 2-D signals, such as images, extra components are included in this simulation. As the image sensor produces 2-D data, the transmitter has to have a data conversion and reshape operations to prepare the captured signal for the transmitter.

The main challenge is to deal with floating point data that comes from the reshape block. Therefore, a special floating-point re-quantization operation has been introduced in the proposed simulation. Re-quantizing the integer and the floating point parts separately has proven the ability of retrieving data efficiently [35]. This re-quantizer is programmed in a separate m-file function and injected into the simulation model as a stand-alone block.

Converting the 2-D to 1-D signal may cause delays and conflicts with the processing operation in the simulation. Hence, buffering is required to maintain the data flow synchronized between the data source and the subsequent blocks in the simulation. After buffering the image, a 1-D signal will pass through the same stages as will be discussed in the following subsections. Fig. 3 below shows the MWSN transmitter simulation blocks.

A. Signal Encoder

The first step is to encode the multimedia data using a high quality encoder. The encoder concatenates the signal s(t) , the delayed version of the signal s(t-1) and 16-bit binary cyclic encoder for s(t) . The encoded information will be the payload for the transmitted packet. Fig. 4 illustrates the encoded binary signal, which will be the frame payload later. Buffering is necessary to maintain the bits order for the following parts of information:

1) The original samples of the signal,
2) The upsampled version of the original samples
3) The 16-bit cyclic code.

The Zero-Order Hold (ZOH) block is used to keep those

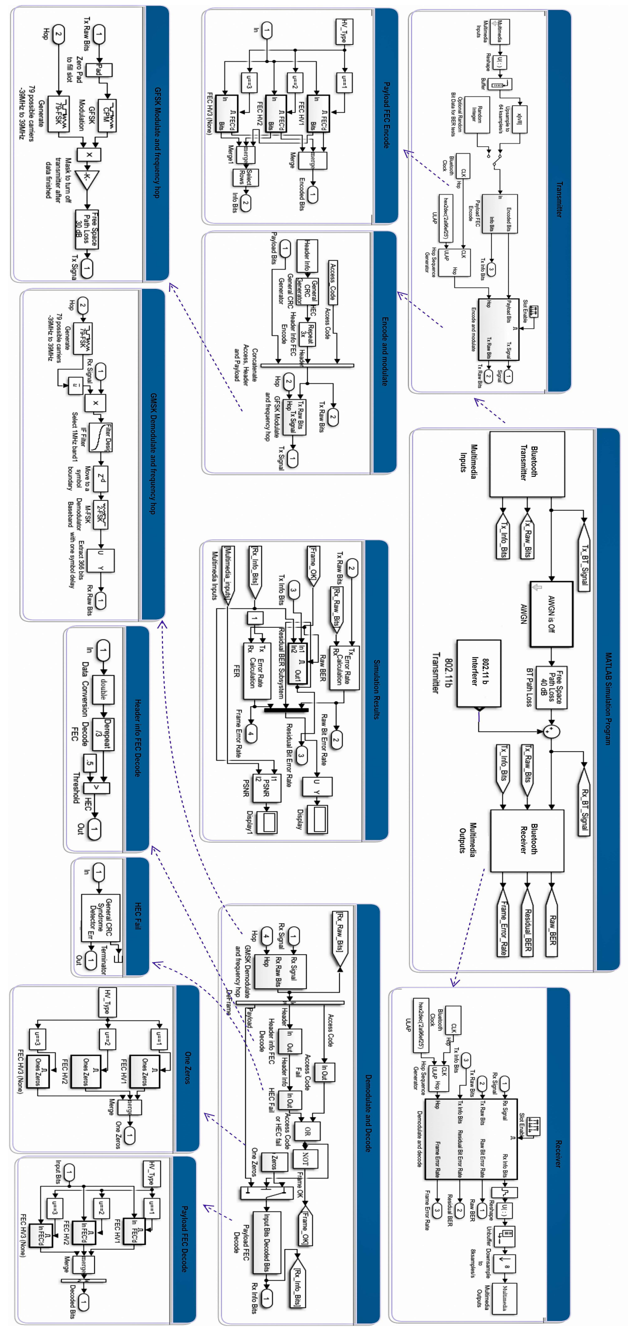


Figure 2. The Proposed Multimedia MATLAB/Simulink general simulation design.

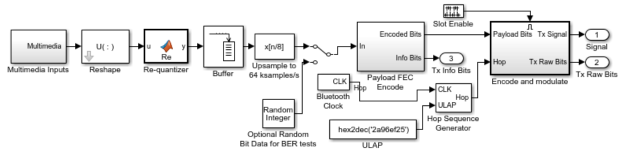


Figure 3. MWSN transmitter simulation blocks.

parts synchronized before merge operation, by deleting the buffering effect, as the merge block may take a CPU clock cycle at running phase. The 16-Bit binary cyclic code is one of the linear code types. Adding a binary cyclic code is useful to detect errors that are added by the channel, and it is also computationally efficient.A 16-bit binary cyclic code can use any polynomial function, for example:

$$p(x) = x^0 + x^2 + x^4 + x^5 + \dots \quad (1)$$

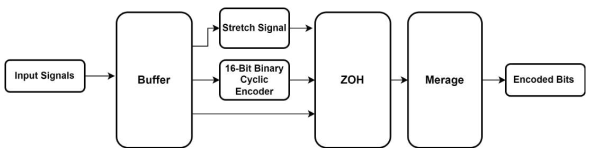


Figure 4. The encoded bits.

Equation 1 means that the binary polynomial is [1 1 0 1 0 1], and the CRC is obtained by XOR division by information bits. Note that this polynomial can be modified in the proposed simulation. The efficiency of cyclic code comes from the fact that its hardware implementation is based on using logic gates and shift registers to divide and multiply information bits [36].

The encoded bits in Fig. 4 are sent through the channel to the receiver, and hence may suffer from corruption. The information bits frame with the same size of the encoded bits frames are formed for error calculation purposes. That form imitates the received payload when there is no error or no channel effect, and it is generated as shown in Fig. 5.

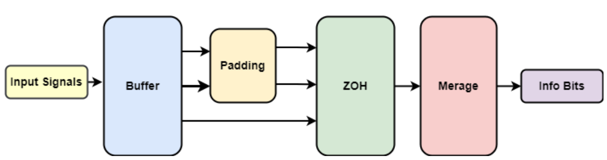


Figure 5. Forming information bits.

B. CRC and Packet Forming

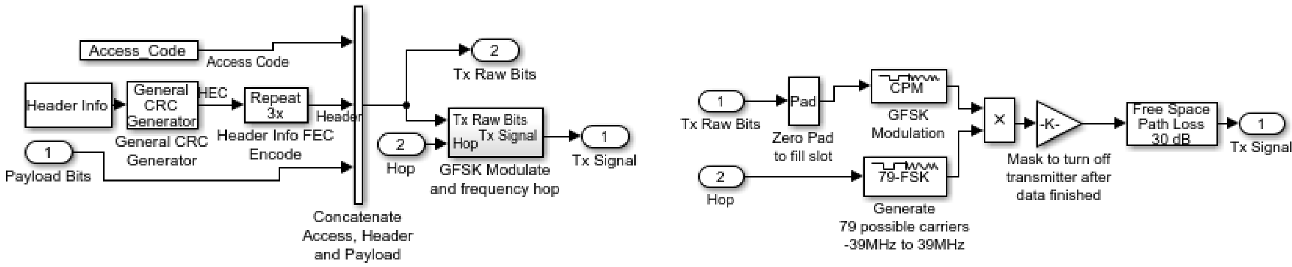
In order to provide proper communication between the sender and the receiver, the transferred data must be divided into blocks, and then extra bits are added to create packets. This process is called framing and coding as explained in section 4-A. For example, the Bluetooth header consists of the slave address, packet type, flow control information, automatic repeat request flag and the sequence number of the current packet [37]. These extra bits are used to synchronize the signal between transmitter and the receiver, and detect any noise, interference, and non-linearity. When the receiver determines errors, error correction should be employed to recover the lost data. This correction can be done in two methods. The first one is simply by asking the sender to repeat the transmission, this technique is called automatic repeat request. The second method is called forward error correction detection(FEC). This technique analyzes the frames by examining the extra bits to detect the errors and tries to correct data [38]. Cyclic redundancy check (CRC) is one of the most widely used algorithms to create codewords for error detection and correction. The codeword is computed using some mathematical operations on the data[39],[40].

C. Modulation

Gaussian Frequency Shift Keying (GFSK) has been applied to modulate the transmitted data. GFSK modulator is very similar to FSK except for passing the transmitted signal through a Gaussian filter. The Gaussian filter is used to smooth the signal and limit the bandwidth spectrum [41]. Decreasing the spectral bandwidth using this filter reduces the bit error rate (BER), where the Gaussian filter balances the trade off between the bandwidth spectrum and the BER [42], [43]. The proposed simulation uses continuous phase GFSK modulation (CPM). A radio channel is used to modulate the signal using Gaussian frequency shift keying (GFSK). To overcome the interference throughout the channel, frequency hopping technique is applied in order to decrease the interference with other devices. The time domain of the transmitted signal is divided by the sender into multiple slots where the modulation frequency is changed at each time [44]. In the M-FSK frequency hopping technique, M represents a different number of frequencies which are used to modulate the signal. The M-FSK modulator is used widely in short range applications [45]. Fig. 6(a) shows how modulation is implemented in the suggested simulation. Firstly, the payload, FEC header and the access code bits are concatenated. Then, the resultant raw bits are modulated using CPM-GFSK as shown in Fig. 6(b).

5. The Channel

For more realistic channel imitation, the proposed simulation allows the user to choose different types and levels of corruption that are usually generated by the medium channel. First, we add a white Gaussian noise. Then, the modulated signal is attenuated to assess the effect of the channel path losses. Finally, an interference signal is added to simulate the transmitted signals through channels. Additive white Gaussian noise(AWGN) block is used to add noise to the transmitted data. The amount of AWGN is controlled by the Signal to noise ratio (SNR) parameter. In the suggested simulation, the amount of noise is altered by changing the SNR from 1 to 5, 10, 15, and 20.



(a) (b)

Figure 6. Modulation Block (a) The general modulation scheme (b) GFSK Modulation inner block details.

The second type of corruption is the path loss attenuation, which is mainly affected by the atmosphere. In general, the distance between the transmitter and the receiver is the critical factor which determines the amount of the attenuation of the transmitted signal. Therefore, in the path loss MATLAB block, the amount of attenuation is controlled by the setting distance between the transmitter and the receiver.

Finally, the distortion of the transmitted data, due to signal interference, is simulated using an 802.11b interference block. In this simulation, the effects of AWGN with or without the interference are analyzed. Meanwhile, the path loss is always added to the channel because it is inevitable.

6. The MWSN Receiver Simulation

The receiver has to reciprocate the modulation procedure to extract the transmitted information. To make the simulation meaningful, the MWSN receiver measures the quality of the received signal as compared to the transmitted signal, as shown Fig. 7.

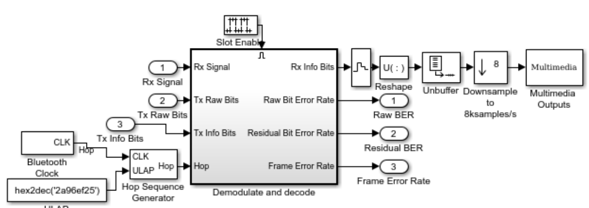


Figure 7. The MWSN receiver simulation blocks.

The "Demodulate and Decode" subsystem block consists of several blocks and logical functions to extract the received information bits from the received signal Rx, as demonstrated in Fig. 8. he function of the "Demodulate and Decode" block can be summarized as follows:

1) GMSK Demodulator that converts the signal to binary data for each frame.
2) Deframe block is used to separate the header and the information parts from the frame.
3) The FEC logical processes that are responsible for accepting or rejecting the frame.
4) Payload decoder which will extract the received information bits from the accepted frame.

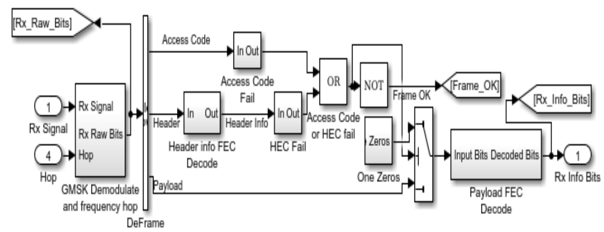


Figure 8. The Demodulate and Decode subsystem.

Regarding the GMSK demodulator subsystem, the first step is to keep the received signal synchronized with the transmitter. The hop sequence is generated for synchronization in the proposed model. The received signal is multiplied by the carrier signal, which is generated by the M-levels frequency shift keying (MFSK). It is impractical to implement the coherent M-FSK receiver as it requires complicated correlation processes [46]. Therefore, the non-coherent M-FSK demodulator is used, which does not estimate the transmitted carrier phase. A Butterworth low pass filter moves the signal spectrum from the pass band to the base band. To extract the binary information from the baseband signal, a 2-FSK demodulator has been used. Fig.9 shows the contents of GMSK demodulate and frequency hop operations.

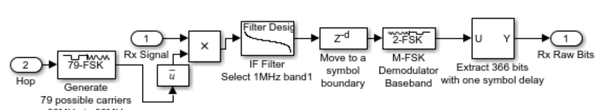


Figure 9. GMSK demodulator and frequency hop at the receiver.

The raw information bits are split into three groups, the header, the payload and the access code. The header bits are processed at FEC decode, in which the CRC detector exits.

If the result of dividing the packet bits by the generator polynomial is zero, the packet is accepted, otherwise it is deleted at the receiver. The detected payload is then decoded back to its original form by reversing the steps discussed in section 4-A. The decoded bits are un-buffered by converting the frame to scalar samples at a higher rate. The information bits are downsampled to be suitable for multimedia presentation at the user end. Finally, the de-quantizer operation is added to the system to reverse the re-quantizer effect at the transmitter. When it comes to the 2-D MWSN receiver, preparing the data for the front-end user is important. The information bits have no meaning for the receiving end user, until it is reconstructed in a similar form as it was transmitted. Hence, the image is reconstructed by converting the 1-D to 2-D data in row-by-row order.

7. The Results

The proposed multimedia WSN simulation is tested under different circumstances and with different types of images, as well as random and voice signals. The test images were carefully selected to represent the possible usage of the MWSN. Those images were an X-ray medical for COVID-19 [47], a car speed monitoring [48], and a plant leaf image [49]. The experiments were performed to test the noise and interference effect to measure the images quality under three different scenarios, which are:

1) The channel adds noise only.
2) The channel adds noise and has WiFi interference 802.11b.
3) Only interference noise exists over the channel.

Using the proposed MWSN simulator, the transmitted and the received multimedia signals, which have been degraded by different channels, are shown in tables II and III. Table II lists the simulation results when the channel corrupts signals at different levels of signal to noise ratio (SNR), while no interference exists.

TABLE II. Noise effect on test signals

Case No	Tested Input Signal	Noise only				
		SNR=1	SNR=5	SNR=10	SNR=15	SNR=20
1						
2						
3						
4						
5						

The simulated square signal(Case 1) is completely corrupted at SNR = 1, the noise effect gradually diminishes for higher SNRs, as depicted in the first row. The second case in the tables shows the simulation results for sample audio signal from Noisy TIMIT corpus from [50]. Different types of images which are tested under the same circumstances are also included in the table. Table III, shows the signal quality for the second scenario, when the channel adds noise and the WiFi interference at the same time.

TABLE III. Noise and interference effect on test signals

Case No	Tested Input Signal	Noise and Interference Effect				
		SNR = 1	SNR = 5	SNR = 10	SNR = 15	SNR = 20
1						
2						
3						
4						
5						

In both scenarios, the received signal quality is accepted for SNR higher than 15 dB, with a distorted signal at 10 dB, and a blank image at lower SNR values. The system failed at SNR values less than 10 dB as expected as a result of a very tough channel behavior. For some computer vision applications, the 10 dB SNR will not prevent the recognizer from doing their job. However, other sensitive applications may not accept this level of quality.

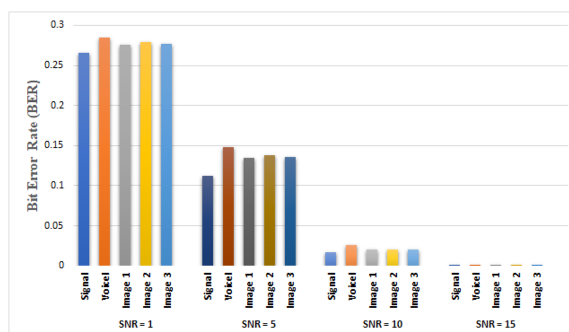


Figure 10. SNR vs. BER (Noise Only).

The peak signal to noise ratio (PSNR) between the transmitted and the received images is measured using the simulation. The relationship between the bit error rate (BER) and the PSNR is depicted in Fig. 10. On the other hand, The PSNR versus the noise level relationship is shown in Fig. 11, where the interference is ignored. The outcome of the proposed MWSN, as those presented in this section, can help researchers to understand the effect of channel corruption on the 1-D and 2-D signal. Besides, this analysis helps the designer to understand and assess the performance of the WSN for different types of channels.

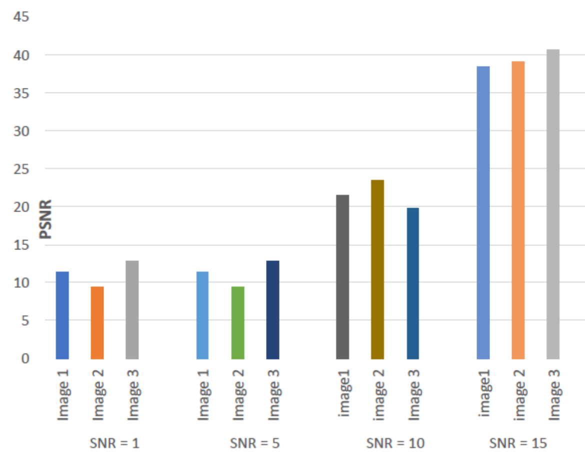


Figure 11. SNR vs. PSNR (Noise Only).

Additionally, another scenario is examined where the channel exposes noise and interference at the same time, the results of this simulation are depicted in Figures 12 and 13. Furthermore, The interference effect is considered solely where the PSNR, BER and FER are measured for the tested 2-D image signals only, and the results are shown in Fig 14. Finally, 1-D signals are tested with different levels of interference power and BER is computed for each level.

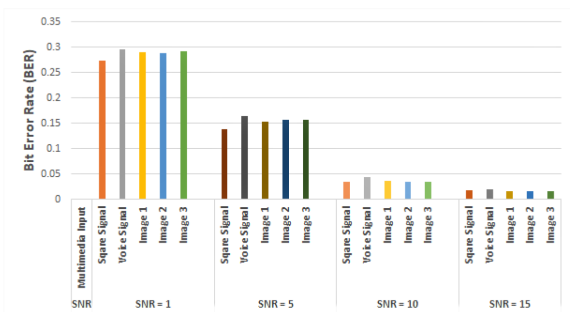


Figure 12. SNR vs. BER for Noise and Interference.

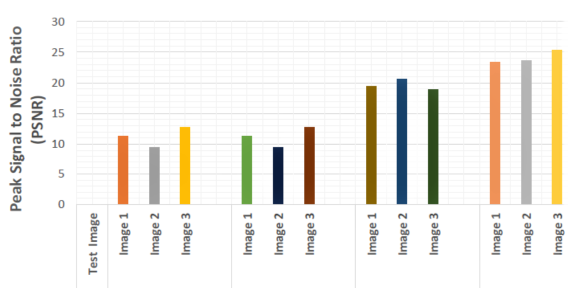


Figure 13. SNR vs. PSNR for Noise and Interference.

The results are shown in Fig. 15 for audio and deterministic signals, which depicts the BER versus normalized

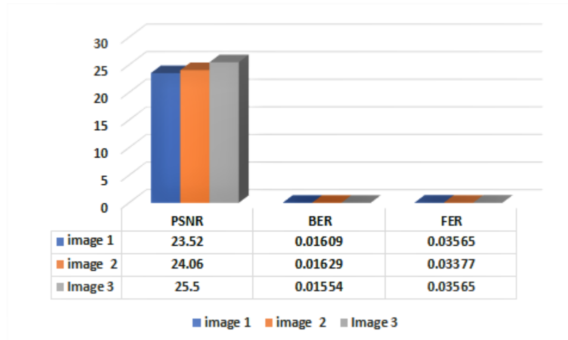
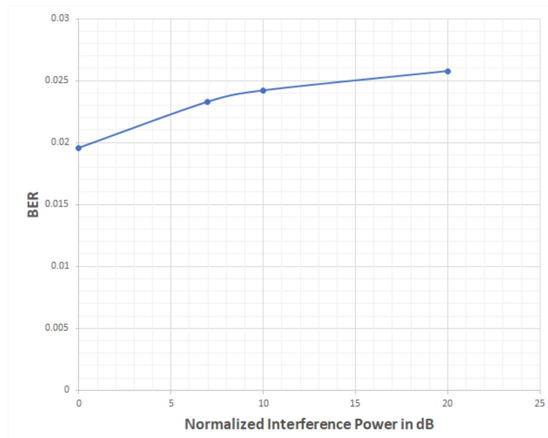
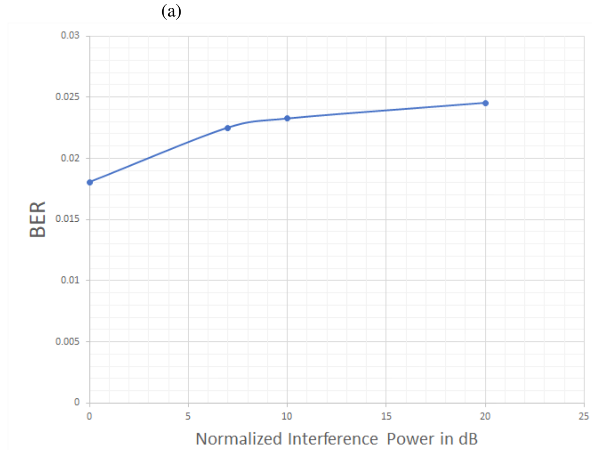


Figure 14. System performance for different interference power for 2-D image signals.



(a)



(b)

Figure 15. System performance for different interference levels, (a) the BER for audio signal (b) the BER for the square wave signal.

power in each case. It is noticed that the BER increases rapidly between 0 and 5 dB of normalized interference power. However, it seems that the BER slightly increases for interference levels higher than 10 dB for both deterministic and speech signals.

8. Conclusions

The proposed Matlab/Simulink model in this paper deals with multi-dimensional signals with high scalability. Multi-dimensional signals' behavior has been inspected with different noise and interference levels using the proposed multimedia WSN (MWSN) simulation model. In this case, the WSN that uses 1-D and/or 2-D signals can be analyzed and assessed, to predict the proper solution before applying it in real life. It will benefit the WSN and IoT designers to reduce costs, time and avoid faults. The main challenge was to deal with floating-point data as a result of filters in the transmitter/receiver that damage the received image signals during conversion operations, which produce floating-point numbers that lead to uncertain quality measures. Therefore, a special floating point re-quantization operation had been introduced in the proposed simulation. Besides, adding more buffers to keep signal synchronization to avoid delays and conflicts. The experimental results have shown that the designed MWSN is a flexible tool for modeling the physical layer of the WSN for 1-D and 2-D signals. Additionally the MWSN supports almost all the crucial parameters to be adjustable. The relationship between the channel noise levels and the received signal quality proved the validity of the proposed modeling approach. Nowadays, the WSN applications, such as telemedicine, agriculture, traffic control, demands using muli-dimensional signals. Therefore,a multimedia model that deals with such signals has been proposed and tested in our experiments. The MATLAB/SIMULINK scalability supports the expansion of the current model to build different topologies. Moreover, external hardware, like signal generators , sensors and external cloud services can be connected into the MWSN model. The proposed model in this paper has a potential to be extended for different networking technologies and more applications.

References

[1] A. Sharma, P. K. Singh, and Y. Kumar, "An integrated fire detection system using iot and image processing technique for smart cities," *Sustainable Cities and Society*, vol. 61, p. 102332, 2020.

[2] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, "Applications of wireless sensor networks: an up-to-date survey," *Applied System Innovation*, vol. 3, no. 1, p. 14, 2020.

[3] A. Doublali, A. Jilbab, C. Bojji, and R. Idchabani, "Smart wall by wireless sensor network toward building energy optimization," in *E3S Web of Conferences*, vol. 336. EDP Sciences, 2022, p. 00032.

[4] R. Sharma, V. Vashisht, and U. Singh, "Modelling and simulation frameworks for wireless sensor networks: a comparative study," *IET Wireless Sensor Systems*, vol. 10, no. 5, pp. 181–197, 2020.

[5] P. S. Katkar and D. V. R. Ghorpade, "Comparative study of network simulator: Ns2 and ns3," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 6, no. 3, pp. 608–612, 2016.

[6] N. Hrovatin, A. Tošić, and J. Vičič, "Ppwsim: Privacy preserving wireless sensor network simulator," *SoftwareX*, vol. 18, p. 101067, 2022.

[7] I. F. Kurniawan and R. Bisma, "Multiple sensing application on wireless sensor network simulation using ns3," in *Journal of Physics: Conference Series*, vol. 947, no. 1. IOP Publishing, 2018, p. 012011.

[8] A. Kurniawan, P. Kristalina, and M. Z. S. Hadi, "Performance analysis of routing protocols aodv, olsr and dsdv on manet using ns3," in *2020 International Electronics Symposium (IES)*. IEEE, 2020, pp. 199–206.

[9] C. Shastry *et al.*, "Design of wsn model with ns2 for animal tracking and monitoring," *Procedia Computer Science*, vol. 218, pp. 2563–2574, 2023.

[10] K. Swaminathan, V. Ravindran, R. Ponraj, and R. Satheesh, "A smart energy optimization and collision avoidance routing strategy for iot systems in the wsn domain," in *Applied Computational Technologies: Proceedings of ICCET 2022*. Springer, 2022, pp. 655–663.

[11] P. Chithaluru, L. Jena, D. Singh, and K. Ravi Teja, "An adaptive fuzzy-based clustering model for healthcare wireless sensor networks," in *Ambient Intelligence in Health Care: Proceedings of ICAIHC 2022*. Springer, 2022, pp. 1–10.

[12] P. K. Sharma and U. Modani, "Implementation of secure communication framework for wireless sensor network," in *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 2*. Springer, 2022, pp. 595–604.

[13] M. F. B. Lakarani, A. B. A. B. Sajak, M. N. B. Ismail, and N. B. Allias, "Investigation of the impact of routing protocol in green iot with netsim," in *2022 1st International Conference on Information System & Information Technology (ICISIT)*. IEEE, 2022, pp. 421–424.

[14] B. Priyeash and J. Thyagarajan, "Performance evaluation and comparison analysis of aodv and rpl using netsim in low power, lossy networks," in *Futuristic Communication and Network Technologies: Select Proceedings of VICFCNT 2020*. Springer, 2022, pp. 11–22.

[15] T. Al-Khraishi and M. Quwaider, "Performance evaluation and enhancement of vlan via wireless networks using opnet modeler," *arXiv preprint arXiv:2007.06997*, 2020.

[16] H. S. Obaid, "Wireless network behaviour during jamming attacks: Simulation using opnet," in *Journal of Physics: Conference Series*, vol. 1530, no. 1. IOP Publishing, 2020, p. 012009.

[17] D. Chen, Y. Song, and T. Yang, "Wireless network simulation platform based on opnet and simulink," in *Recent Advances in Sustainable Energy and Intelligent Systems: 7th International Conference on Life System Modeling and Simulation, LSMS 2021 and 7th International Conference on Intelligent Computing for Sustainable Energy and Environment, ICSEE 2021, Hangzhou, China, October 22–24, 2021, Proceedings, Part II 7*. Springer, 2021, pp. 408–417.

[18] N. N. Alfaisaly, S. Q. Naeem, and A. H. Neama, "Enhancement of wimax networks using opnet modeler platform," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 3, pp. 1510–1519, 2021.

[19] V. Suryawanshi and M. Khandekar, "Design and development of wireless sensor network (wsn) for water quality monitoring using zigbee," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018, pp. 862–865.

[20] W. Liu, H. Wang, L. Yu, M. Leach, and F. Ma, “Wireless sensor network traffic modeling and anomaly simulation based on opnet.” *International Journal of Design, Analysis & Tools for Integrated Circuits & Systems*, vol. 7, no. 1, 2018.

[21] A. M. Almajidi and V. Pawar, “A new system model for sensor node validation by using opnet,” *Wireless Personal Communications*, vol. 108, pp. 2389–2401, 2019.

[22] Y. Qin, “Research on opnet energy simulation application in wireless sensor network,” in *Journal of Physics: Conference Series*, vol. 1952, no. 4. IOP Publishing, 2021, p. 042022.

[23] K. Obelovska and I. Danych, “Adoption of the omnet++ simulator for the computer networks learning: A case study in csma schemes,” in *Advances in Artificial Systems for Logistics Engineering*. Springer, 2022, pp. 234–243.

[24] R. Torres-Carrión, C. Gonzaga-Acaro, and H. Torres-Carrión, “Design and simulation in omnet++ of a wireless sensor network for rural exploitation zones,” in *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*. IEEE, 2021, pp. 1–6.

[25] P. Ledesma, D. Gotti, and H. Amaris, “Co-simulation platform for interconnected power systems and communication networks based on pss/e and omnet++,” *Computers and Electrical Engineering*, vol. 101, p. 108092, 2022.

[26] H. A. Babaeer and S. A. Al-Ahmadi, “Efficient and secure data transmission and sinkhole detection in a multi-clustering wireless sensor network based on homomorphic encryption and watermarking,” *IEEE Access*, vol. 8, pp. 92 098–92 109, 2020.

[27] Z. Albayrak, M. Hatem, and M. Çakmak, “Performance evaluation of wbans mac protocols in different dbm and omnet++,” *Journal of Intelligent Systems: Theory and Applications*, vol. 4, no. 1, pp. 1–7, 2021.

[28] M. K. Baazaoui, I. Ketata, G. Fersi, A. Fakhfakh, and F. Derbel, “Implementation of rssi module in omnet++ for investigation of wsn simulations based on real environmental conditions.” in *SENSORNETS*, 2022, pp. 281–287.

[29] R. P. de Medeiros, J. M. M. Villanueva, and E. C. T. de Macedo, “Lifetime increase for wireless sensor networks using cellular learning automata,” *Wireless Personal Communications*, vol. 123, no. 4, pp. 3413–3432, 2022.

[30] A. Prajapati and B. Banerjee, “Simulating distributed wireless sensor networks for edge-ai,” in *2022 5th Conference on Cloud and Internet of Things (CIoT)*. IEEE, 2022, pp. 75–82.

[31] M. L. Rajaram, E. Kougianos, S. P. Mohanty, and U. Choppali, “Wireless sensor network simulation frameworks: A tutorial review: Matlab/simulink bests the rest,” *IEEE Consumer Electronics Magazine*, vol. 5, no. 2, pp. 63–69, 2016.

[32] G. S. Beula and P. Rathika, “Zigbee transceiver design for smart grid home area network using matlab simulink,” in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*. IEEE, 2020, pp. 1–5.

[33] A. Pang, F. Chao, H. Zhou, and J. Zhang, “The method of data collection based on multiple mobile nodes for wireless sensor network,” *IEEE Access*, vol. 8, pp. 14 704–14 713, 2020.

[34] M. Karthiga and V. Santhi, “A wireless sensor network for remote detection of arrhythmias using convolutional neural network,” *Wireless Networks*, pp. 1–12, 2022.

[35] A. A. M. DAWOOD, A. S. ABDULAZIZ, and A. J. MOHAMMED, “Rlc-based image compression using wavelet decomposition with zero-setting of unnecessary sub-bands,” *Journal of Engineering Science and Technology*, vol. 17, no. 1, pp. 0391–0403, 2022.

[36] P. Ivaniš and D. Drajić, *Information Theory and Coding-Solved Problems*. Springer, 2017.

[37] M. Cäsar, T. Pawelke, J. Steffan, and G. Terhorst, “A survey on bluetooth low energy security and privacy,” *Computer Networks*, vol. 205, p. 108712, 2022.

[38] D. H. Morais and Morais, *5G and Beyond Wireless Transport Technologies*. Springer, 2021.

[39] L. Gaus, M. Schwarz, and J. Börcsök, “An integrated procedure for dynamic generation of the crc in view of safe wireless communication,” in *International Congress and Workshop on Industrial AI 2021*. Springer, 2022, pp. 256–265.

[40] Z.-N. Li, M. S. Drew, J. Liu, Z.-N. Li, M. S. Drew, and J. Liu, “Introduction to multimedia,” *Fundamentals of Multimedia*, pp. 3–26, 2021.

[41] D. H. Morais and D. H. Morais, “5g transport payload: Ethernet-based packet-switched data,” *5G and Beyond Wireless Transport Technologies: Enabling Backhaul, Midhaul, and Fronthaul*, pp. 19–31, 2021.

[42] K. S. Mohamed, *Bluetooth 5.0 Modem Design for IoT Devices*. Springer, 2022.

[43] P. Supanakoon and S. Promwong, “Evaluation of distance error with bluetooth low energy transmission model for indoor positioning,” *Journal of Mobile Multimedia*, pp. 707–722, 2021.

[44] N. Benvenuto, G. Cherubini, and S. Tomasin, *Algorithms for communications systems and their applications*. John Wiley & Sons, 2021.

[45] S. Sachan, R. Sharma, and A. Sehgal, “Energy efficient scheme for better connectivity in sustainable mobile wireless sensor networks,” *Sustainable Computing: Informatics and Systems*, vol. 30, p. 100504, 2021.

[46] J. Liu, Y. Yang, and H. Yu, “The simulation of a complete digital communication system,” in *International Conference on Intelligent Systems, Communications, and Computer Networks (ISCCN 2022)*, vol. 12332. SPIE, 2022, pp. 283–293.

[47] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. A. Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan *et al.*, “Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images,” *Computers in biology and medicine*, vol. 132, p. 104319, 2021.

[48] J. G. Proakis, M. Salehi, and G. Bauch, *Contemporary communication systems using MATLAB*. Cengage Learning, 2012.

[49] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” *Frontiers in plant science*, vol. 7, p. 1419, 2016.

[50] A. S. Abdulaziz and V. Z. Këpuska, "Noisy TIMIT Speech LDC2017S04," Web Download. Philadelphia: Linguistic Data Consortium, March 2017, ISBN: 1-58563-793-9.

Akram Abdul-Mawjood Dawood A faculty member of the Computer Engineering department at Mosul university/Nineveh, Iraq.He holds B.Sc. in Computer Engineering, college of Engineering, University of Mosul in Iraq in 2002. He received his M.Sc. in computer engineering from Mosul University in 2008. He published many papers and got a promotion in 2014 in Architecture and computer communication. the research field are Image processing , Artificial intelligence,Digital signal processing Machine learning, Computer communication and computer networks.

Dr. Azhar Sabah Abdulaziz is the head of Software Department, College of Information Technology Ninevah University. He holds B.Sc. in Computer Engineering, college of Engineering, University of Mosul in Iraq in 2002. He was honored with two scholarships from the Iraqi government to complete his post-graduation studies. Firstly, the M.Sc. in University of Newcastle in the UK in 2009 and the second scholarship was for the Ph.D. in Computer Engineering from Florida Institute of Technology in the USA in 2018. He is currently interested in artificial intelligence in signal processing and digital communication.

Amar Daood A faculty member of the Computer Engineering department at Mosul university/Nineveh, Iraq. He received his M.Sc. in computer engineering from Mosul University in 2008. He completed his Ph.D. in 2018 in Computer engineering from Florida institute of technology Florida USA. Dr Daood's current areas of research are Computer Vision, Artificial intelligence, Machine learning, Computer Graphics, Real time system and Parallel Computing.

Qutaiba Ibrahim Ali was born in Mosul City, Iraq in 1974. He received the BS and MS degrees from the Department of Electrical Engineering, University of Mosul, Iraq, in 1996 and 1999, respectively. He received his PhD degree (with honor) from the Computer Engineering Department, University of Mosul, Iraq, in 2006. Since 2000, he has been with the Department of Computer Engineering, Mosul University, Mosul, Iraq, where he is currently a full professor. His research interests include computer networks analysis and design, embedded network devices, and network security. He instructed many topics (for post and undergraduate stage) in computer engineering field during the last 20 years and has more than 80 different publications in many ISI indexed journals and conferences. He acquires many awards and appreciations from different parties for excellent teaching and extra scientific research efforts. Also, he was invited to join many respectable scientific organizations such as IEEE, IENG ASTF, WASET and many others. He was participating as technical committee member in more than 55 IEEE international conferences and joined the editorial board of 10 scientific international journals.