LAUTECH Journal of Computing and Informatics (LAUJCI) – ISSN: 2714-4194
Volume 2 No.2, Sept 2021 – www.laujci.lautech.edu.ng

Development of Turbo Code Error Detection and Correction Scheme for Wireless Telemedicine Video Transmission

Olayinka O.O.
Department of Computer Engineering, LAUTECH
Ogbomoso, Nigeria.
olayinkaoyeronke@gmail.com

Emuoyibofarhe O. J.
Department of Information System, LAUTECH.
Ogbomoso, Nigeria
eojustice@gmail.com

Oladosu J.B.
Department of Computer Engineering, LAUTECH.
Ogbomoso, Nigeria.
johnoladosu@gmail.com

ABSTRACT
Transmission of medical images and videos to a distant location over a wireless network for a proper diagnosis of the patient is a core aspect of telemedicine. During data transmission over the wireless communication channel, noise and other impairments are introduced into data and this causes error in the transmitted data. Hence, there is a need for a method to detect and correct error which may lead to an erroneous diagnosis. Turbo codes happens to be the earliest error-correcting codes with the intention of establishing a dependable communications near the channel capacity with basically possible hardware. It has an excellent error correcting capabilities, which make it appropriate for many internet communications technology. This paper has come up with an effective and efficient method to detect and correct error encountered during the transmission of telemedicine video over the wireless channel with the use of a parallel concatenated Turbo code error detection and correction scheme. A MATLAB Simulation was carried out to investigate and demonstrate the performance of the proposed system. The performance of the developed system was taken at different ranges of SNR with BER, PSNR, MSE and the processing time of the decoders. The developed system was compared with when turbo code is not applied during transmission. The results of the simulation shows a better performance when turbo code was applied compared to when it was not applied.

Key words: Telemedicine, Error detection and correction, Wireless Communication system, Turbo Code, MATLAB, SNR, BER, PSNR and **MSE**.

1. INTRODUCTION

Communication between human being has a long and varied history with the use of many different techniques and methods. These techniques and methods have been determined by circumstances and the available technology. There has been an incredible increase in the trends of digital over the years. Information is represented as a sequence of bits in digital communication system and the processing is done in the digital domain (Wang 2013). Information Communication and Technology (ICT) emphasis the functions of telecommunication, this entails communication devices and applications such as television, cellular phones, radio, fax machine, printer, scanner, camera, satellite systems, computer hardware and software, network hardware and software and so on. This is widely utilized by specialized businesses and several industries such as engineering, banking, education, manufacturing, telecoms and health care. This has given the industry's ability to access, store, retrieve, transmit, and manipulate information within and outside their location by the aid of physical wires or radio links (Lathi and Ding 2009). The improvement in ICT has brought about the development of high-speed networks that are able to transmit high quality, full-motion video by combining the capabilities of both telephone and computer networks into one network (Aderemi et. al., 2013).

The merging of telecommunication and communication technologies is the major backbone of telemedicine care services. Telemedicine can simply be defined as a process whereby two or more health giver and receiver discuss over the telephone or using the satellite technology to broadcast a consultation from different location using video conferencing equipments (Dejan et. al., 2011). Some of the advantages of telemedicine are the capability to provide timely means of having access to experienced health care givers that are far with the help of telecommunication and information technologies, irrespective of the patient's location. Video based applications such as multimedia messaging, video streaming, VoIP, teleconferencing and telemedicine video have widely utilized Wireless Local Area Network (WLAN) due to ease of installation and deployment. Some of the challenges that are been encountered by video-related application like telemedicine video during transmission over wireless network include delays, jitter, low throughput and packet loss It is of great importance for the receiver to detect such errors and also be able to correct them otherwise it might have a substantial effect on diagnosis thus making diagnosis less reliable and not efficient (Seyed et. al., 2017). Therefore, a turbo coding system is necessary for the detection and correction of error during telemedicine video transmission over the wireless network to enhance quality of service (QoS) delivered to the patients.

2. LITERATURE REVIEW

The process of transmitting digital messages to devices that are autonomous powered circuitry that exists outside the chassis of a computer system is called data communication. The signal power is directly proportional to the maximum permissible transmission rate of a message and channel noise is inversely proportional to maximum permissible transmission rate (Misha and Mishra 2006). The provision of highest possible transmission rate with least possible noise and lowest possible power is the main target of any communication system. Data transmission occurs when a device like computer is transferring data over a point to point or point to multipoint communication channel from the source/sender to one or multiple destination/recipient device, like a server or computer (Martini and Mazzotti 2006). Data to be transferred could be analog signal, such as a phone call that is been digitized to a bit-stream, using some coding scheme or a digital signal that is in form of discrete signals (Ajala et.al., 2015). Data can be transmitted over the various available channels in different ways: parallel data transmission and serial data transmission. Errors that occur during transmission over the communication channel can be categorized into two main categories which are: single-bit errors and burst errors.

During transmission of data in a communication channel, error might be introduced (Ranjeet 2009). The major source of error in a communication system is noise (Kachienga and Michael 2011). Noise is an electrical signal that is random, undesirable and most time unavoidable. Usually shows itself as extra bits, changed bits or missing bits and is usually caused by natural disturbances or equipment. To reduce the unpredictable effect of noise on data transmitted, some error control schemes need to be implemented (Wang and Zhu 1998). The process of detecting and correcting bits error in digital communication is known as channel coding. This can also be referred to as Forward Error Control (FEC). Channel coding is performed at both sides; that is the transmitter and at the receiver (Huyu et. al., 2011). At the transmitter side; there is an encoder that adds extra bits (parity bits) to the data to be transmitted before modulation takes place. While at the receiver side, there is a decoder that will detect and correct any errors that occur during transmission (Hannah 2014)

The first coding scheme that is near the theoretical limit expressed by Shannon- Hartley law is Turbo code (Ranjeet, 2009). Shannon-Hartley theorem states that if the rate of information of a particular source is not greater than the channel capacity of a given channel, then there will be a coding technique that helps transmission over unreliable channel with random low error rate (Stallings 2016). The Shannon limit is represented as

$$C = Bw * \mathrm{Log}_2(1+ S/N) \qquad (1)$$

C = Maximum Channel Capacity, Bw = Channel Bandwidth, S =Signal Power, N = Noise Power (Emuoyibofarhe and Bardi 2009). The code is near 0.5 dB of the theoretical limit, while conventional code requires up to 3 to 4 dB more than the theoretical limit (Shannon limit 2001). This has enhanced the acceptability of the codes and thus accuracy of the communication system using such codes. The name “turbo" was derived from the similarity it has with turbo engine and the repetitive decoding task interval that separate the two Soft-input Soft-output (SiSo) convolutional decoders at the receiving end. It has some notable characteristics which are strong ability to correct error, satisfactory complexity, and option to maintain distinct sizes of block and rates of code.

3. METHODOLOGY

The methodology adopted in this paper was design and simulation. The architectural overview of the proposed system can be seen in Figure 1. The system comprises of two stations (Medical Expert and Patient stations), some medical diagnosis devices, turbo code encoder and turbo code decoder. Both stations were connected through a WLAN and have a full duplex connection. That is both stations can transmit and receive information at the same time. Ultrasound and Electrocardiogram (ECG) videos were collected from patients using a portable Ultrasound machine and an ECG machine. The video serves as the input to the encoder and can be sent from any of the stations. That is why the two stations were connected to the turbo encoder. The output from the turbo decoder can also go to the both stations this depends on the flow of the communication. Simulations are conducted to compress and convert the diagnostic video to a desired resolution and frame rate. The compressed video was then encoded using a turbo encoder, the modulation used in this paper was BPSK modulation and Additive White Gaussian Noise (AWGN) model used to add noise to the wireless channel. At the end of each iteration, the output of the first turbo decoder was sent back to the second decoder for reconstruction. The quality of the transmitted video using a turbo coding was compared with when turbo code was not applied. The performance of the system was investigated using Bit Error Rate (BER), Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE) and processing time of the decoders.

4. ERROR DETECTION AND CORRECTION USING TURBO CODE

Analysis of a scheme of deploying error correcting codes for Wireless Telemedicine video transmission was done using Parallel concatenation convolutional codes (PCCC) with intuitive turbo decoding, so that the encoding process takes place at the sender station and the decoding process at the receiving end. Parallel concatenated convolutional Turbo codes design comprises of Encoding process, Interleaver design, Puncturing , Decoding process and the decoding Algorithm.

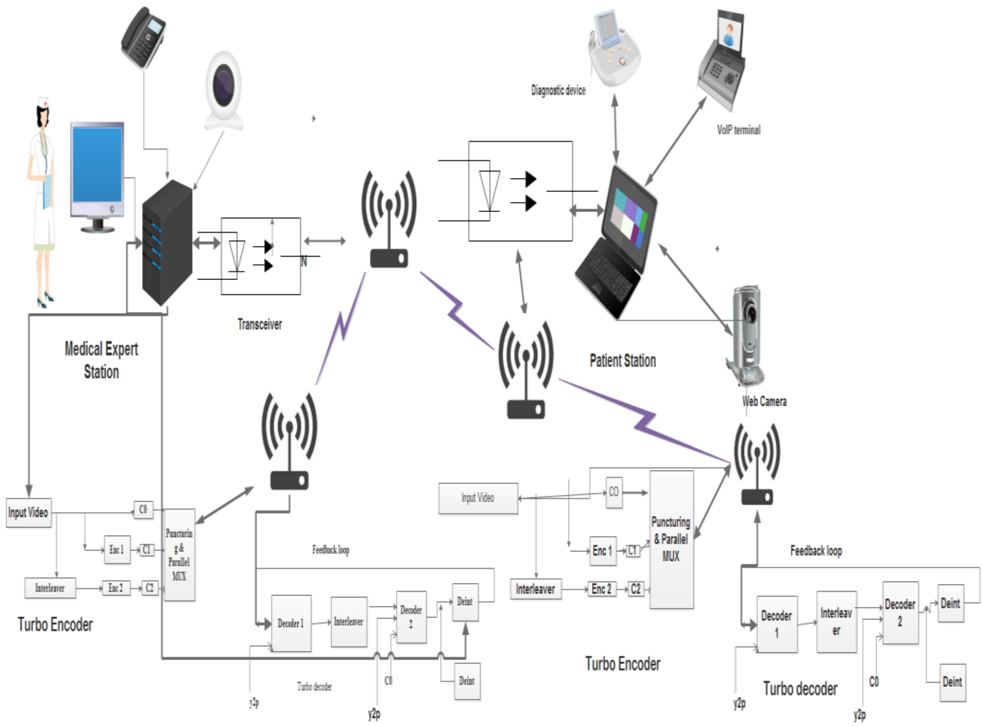


Figure 1: The Architectural overview of the Proposed System

4.1 The Encoding Process

Turbo encoder used in this implementation is the one specified in IEEE802.16e standards that comprises of two Recursive Systematic Convolutional (RSC) codes placed in a Parallel Concatenation and also an interleaver. Videos were captured at a standard acceptable by Digital Imaging and Communications in Medicine (DICOM). The resolution at which videos were captured is Common Intermediate Format (4CIF). The Ultrasound input video image and ECG Video image are shown in Figure 2 and Figure 3 respectively. The resolution has a video sequence of 704×576 resolution and the frame size (in bits) used for this work are (50, 150, 200, 400). Different sizes of the frame are chosen to study their effects on the performance of the system and are chosen randomly.

The video acquired from the patients using a portable ultrasound machine is converted into a binary input data sequence, the data sequence is passed as an input in to the first convolutional encoder C1, and a set of systematic and parity bits are generated. The permuted version of the data bits that is acquired through an interleaver P is encoded by the second encoder C2 to generate another set of systematic and parity bits. The converted video was fed as an input (d_k) into the first encoder ENC1, the video pass through an interleaver before been fed into the second encoder ENC2. The output of the ENC1 and ENC2 were multiplexed and Punctured to give a parity output $X^p{}_k$ The turbo code encoder performance was improved with the help of the interleaver by increasing the minimum distance of the turbo code in such a way that after the error in one dimension has been corrected, the remaining errors in the second dimension would have become correctable error patterns.

Figure 2: Ultrasound Input Video Image

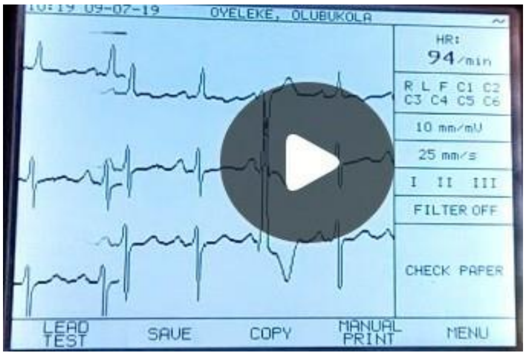


Figure 3: ECG Input Video Image

The block diagram of the turbo encoder is depicted in Figure 4. The structure of the encoder consist of two RSC blocks and a random interleaver.

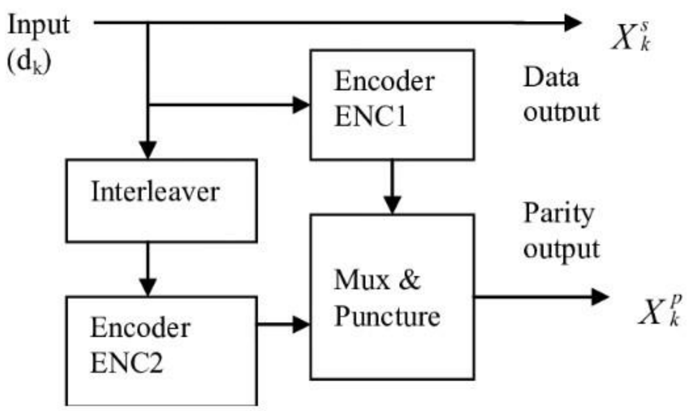


Figure 4: Turbo Encoder Structure

4.2 The Interleaver Design

The performance of turbo code depends majorly on the interleaver design. Interleaver was used between the encoders to increase the rate at which burst errors are being corrected. Types of interleavers available are block interleaver, convolutional interleaver and random interleaver. In block interleaver, the input array is arranged row-wise and read out column-wise and are also formatted in a matrix of d rows and e columns such that N=d*e. But in the random interleaver a block of N input data is read into the interleaver and is read out randomly. In this paper, random interleaver was used. Because the error rate is less compared to the block interleaver. The de-interleaver is available at the decoders as well, so as to perform the decoding correctly. Interleaver design concept is shown in the algorithm:

1. Create a random interleaver.
2. Create possible input data sequences.
3. Determine the resulting code words for every most likely input data sequences.
4. Determine the weight of the code words to find the weight distribution of the code.
5. From the data that are collected, estimate the minimal codeword weight and the amount of code words with that weight.

The algorithm was repeated for a number of times. Comparing the data, the interleaver that have the largest minimal codeword weight and smaller number of code words with that weight was selected.

4.3 Puncturing

The efficient way to increase the code rate of the turbo code system in this paper is to introduce puncturing. Puncturing was introduced after the encoding so as to make available unequal error protection for the encoded information bits. This was done by not allowing the second encoder to send information bits thereby increasing the code rate from $^{1}/_{3}$ to ½. In this paper, a rate compatible punctured code was used, where the two coded bits stream are multiplexed to increase the code rate and also to provide unequal error protection. Figure 5 shows that the selected bit from the output parity stream was punctured to increase the code rate. The odd bits from the first encoder and the even bits from the second encoder were punctured to increase the code rates to ½.

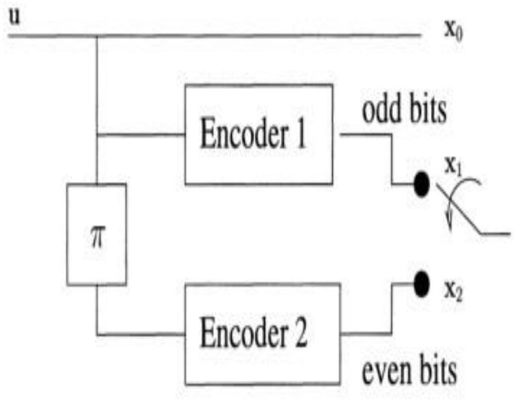


Figure 5: Punctured Turbo Encoder with code rate= ½

4.4 The Decoding Process

The decoding process of the turbo code starts after the encoding process, the whole information bits was put together into a frame and is been modulated and transmitted over the channel. At receiver side, the information bits from the encoders enter the decoders which have been introduced to Gaussian noise as they are transmitted over the channel. Probabilities for systematic information *C0* and coded information *y1p* will be fed into the decoder *D1*, this will provide extrinsic information (a refinement of *C0*) to decoder *D2* (first half-iteration). Then *D2* used this extrinsic information, the probabilities for coded information *y2p* and an interleaved version of *C0* to provide extrinsic information for *D1* (second half-iteration). *D1* and *D2* was implemented using the SOVA (Soft Output Viterbi Algorithm). The received video was continuously refined through successive iterations. The number of iterations that was performed on the data was decided based on the size of the data and type of channel with probability of errors during transmission. In this paper, a total number of six (6) iterations was carried out. The turbo code decoder structure was shown in Figure 6.

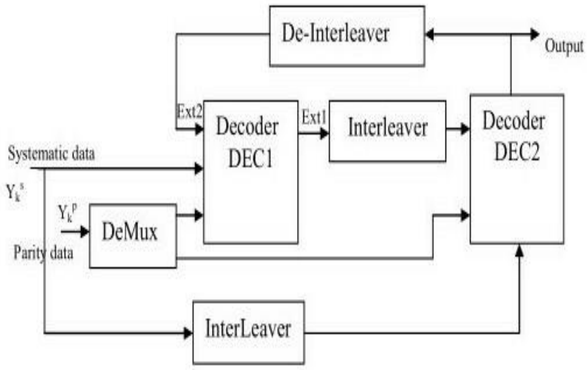


Figure 6: Turbo code Decoder structure

4.5 The Decoding Algorithm

The decoding Algorithm used in this paper was Soft output Viterbi Algorithm (SOVA). SOVA has pass through essential modifications like making use of the path metric value of a prior probability of each input information bits and also generates a soft output. This helps its decision accuracy better than Viterbi Algorithm. The SOVA Algorithm is shown in the following equations:

$$O(u_k / y_k) = O(u_k) + O_c y_{ks} + O_e(u_k) \qquad (2)$$

$O(u_k / y_k)$ represents the soft output of the component decoder, $O_c y_{ks}$ is the value of the received weighted systematic channel, O_c represent the reliability of the channel and $O_e(u_k)$ is the extrinsic value that is produced by the SOVA component decoder.

The path metrics was calculated using Equation 3.

$$M(S^s{}_k) = M(S^s{}_{k-1}) + \frac{1}{2} u_k O(u_k) + O_c / 2 \sum_{i=1}^{n} y_{kl} x_{kl} \text{ ,} \qquad (3)$$

$M(S^s{}_{k-1})$ is the predicted path metric,

n is the length of the message with parity

x_{kl} is encoded bits after the modulation (BPSK).

The first SOVA component decoder output extrinsic information is shown in equation 4.

$$O_{e1}(u_k) = O_1(u_k / y_k) - O_1(u_k) - O_c y_{ks} \qquad (4)$$

The output of the second SOVA component decoder is also an extrinsic information like the first decoder but the $O_c y_{ks}$ is interleaved. The output of the second SOVA component decoder is illustrated as in equation 5:

$$Oe_2(u_k) = O_2(u_k / y_k) - O_2(u_k) - \pi(O_c y_{ks}) \quad (5)$$

The $Oe_2(u_k)$ was deinterleaved, it then return to the feedback of the decoder considering the previous information to the next iteration. This process is repeated for each iteration and finally stops after the sixth number of decoding iteration. The number of iterations used in the decoding process in this work is limited to six iterations so as to reduce latency and energy consumption of the system without losing its performance. At the sixth iteration the soft output is calculated from the output of the second decoder.

4.6 Error Recovery

After a number of decoding iteration, the error introduced to the information bits through the channel has been corrected. The delay, that is the period between the initiation and the occurrence and also energy consumption usually depend on the number of iterations in the process of decoding. Thus, reducing the number of iterations in the process of decoding is very important. The number of decoding iteration performed in this paper was six. Stopping the number of decoding iterations at six saves energy without losing any communication performance. At the end of the sixth iteration the binary bits was converted back to decimal, that is the decoded data. The reconstructed video have recovered from error and is ready for use by the medical expert for analysis and proper diagnosis of the patient.

5. RESULTS AND DISCUSSION

The developed model was tested and simulated on MATLAB R2015a. The turbo encoder, turbo decoder, BPSK modulation and random interleaver were modeled using synchronous dataflow. The results that were obtained from the simulation are discussed here. Some parameters that were used in the development of turbo code were varied during the simulation in order to know their effect on the performance of the turbo code that was being used for the detection and correction of error during the transmission of the telemedicine video. The parameters are: number of decoding iteration, frame size, and code rate. The error metrics used in this work to evaluate the performance of the turbo code system over the AWGN transmission channel are Bit Error Rate (BER), Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE) and processing time of the decoders.

5.1 Bit Error Rate

The result of the simulation was shown in Figure 7. BER simulation was conducted at various SNR Levels with 400bits frame size at the sixth decoding iteration using AWGN to add noise to the communication channel. Simulation was done to compare the result when turbo code was applied and when it was not applied. It can be seen from the graph that the probability that the reconstructed video error when turbo code was used was very low compared with when turbo code was not applied. At 0dB, 1dB, 2db, 3dB, and 4dB SNR, BER results when turbo code was not applied on the reconstructed videos are 0.753, 0.425, 0.289, 0.221 and 0.154 respectively but at 0dB, 1dB, 2dB, 3dB, and 4dB SNR, BER results when turbo was applied BER results are 0.120, 0.028, 0.0004, 1.25×10^{-5} and 3.6×10^{-6} respectively.

The result gotten when turbo code was applied satisfies Shannon theoretical limit for communication and also according to ITU quality scale, examining the video quality objectively, excellent quality video transmission over a communication channel is guaranteed when the bit error probabilities are less than 1×10^{-4}, good quality is in the range of 1×10^{-4} - 4×10^{-4}, satisfactory quality is in the range of 8×10^{-4} - 1×10^{-3}, poor quality is in the range of $8\times10^{-4} – 1\times10^{-3}$, while very bad quality is for any BER $>1\times10^{-3}$. BER values are low with turbo code but high when turbo code was not used. This means that, the probability of the received video to have error is low with turbo code compared with without turbo code.

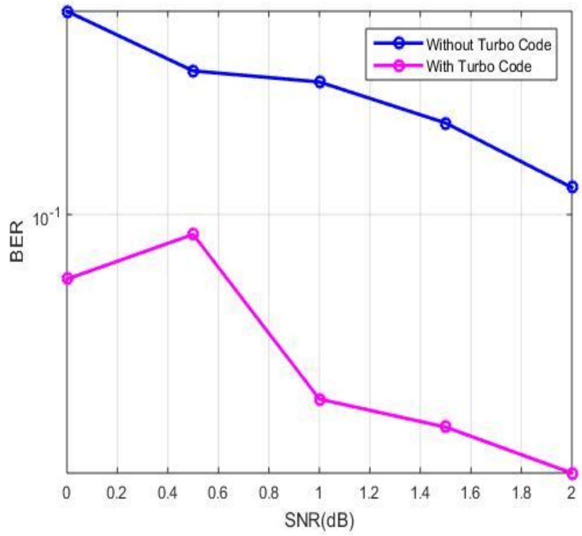


Figure 7: The BER performance of the system with Turbo code and without Turbo code at various SNR Level with Frame Size = 400bits, at the sixth decoding iteration in AWGN Channel.

5.2 Peak Signal-To-Noise Ratio (PSNR)

Simulation was carried out to detect the PSNR of the reconstructed video when turbo code was applied compared with when turbo code was not. The PSNR of the video was achieved by measuring the amount of the greatest value of the signal compared to the power of the distorted noise that corrupts the signals. It can also be expressed as the probability of the peak error and is usually stated in dynamic range in logarithmic decibel scale. The PNSR was calculated using equation 6:

$$PSNR = 10.\log_{10}(MAX^2 / MSE) \qquad (6)$$

MSE stands for the mean square error. MAX is the peak value of original data. Simulation was carried out by introducing bit error into the data by adding AWGN into the channel. The effect on the PSNR when turbo code was applied and when it was not was compared. When the PSNR is high, it shows that the reconstructed video is better and also matches with the original video. This indicates the effectiveness of the decoding algorithm used in this work. Simulation was conducted at various SNR level with different frame rate.

The decoded video quality (in PSNR) of the proposed turbo code scheme with interleaver over AWGN channel result was shown in the Figure 8. From the figure, it can be seen that the curves of the PSNR with turbo code was higher compared with without turbo code. This means that the quality of the decoded video was better when turbo code was applied compared to without when turbo code. When turbo code was applied, the PSNR value of the reconstructed video was high. Simulation was carried out to detect the PSNR value of the reconstructed video at various different frame sizes at SNR 2dB. The result shows that when the frame size is 50bits the PSNR will be 40.48 when turbo code was used but without turbo code it was 38.14. It can be seen that, as the frame size is increasing the PSNR result is getting higher. Also simulation was carried out varying the level of the SNR, when the frame size was 400 at the sixth decoding iteration in AWGN channel and the result was shown in Figure 9. From the figure, the curve of SNR= 4dB is higher than when it is 0dB, 1dB, 2dB and 3dB. Also, the PSNR value with turbo code when the SNR was 0dB gives 43.65, but without turbo code it is 43.15. There is no much difference in the PSNR value when the SNR is low when turbo code was used and without turbo code. That is, the higher the SNR the better the PSNR result when Turbo code was applied.

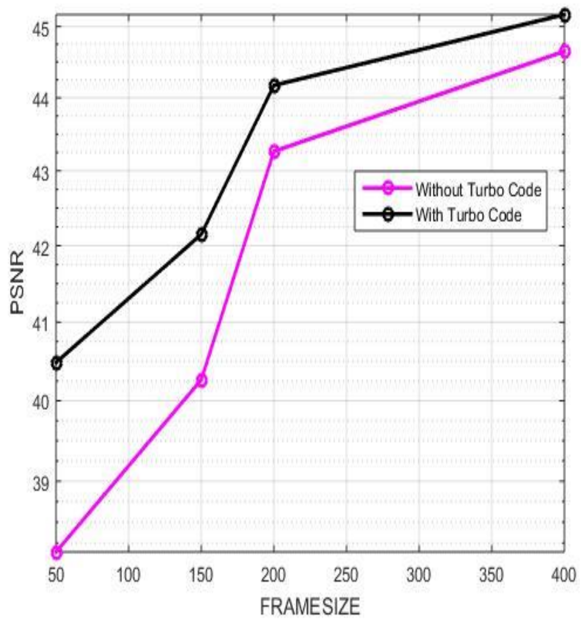


Figure 8: Quality of the reconstructed video (PSNR) when turbo code was applied and without turbo code at varied sizes of frame when the SNR Level is 2dB at the Sixth Decoding Iteration.

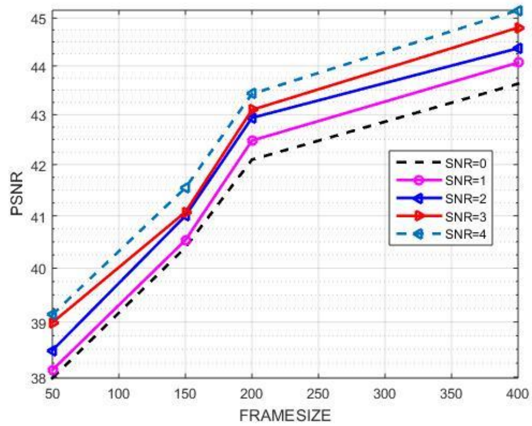


Figure 9: PSNR result when Turbo Code is applied and when not applied at different Frame Sizes and SNR ranges at the sixth decoding iteration with punctured code rate ½.

5.3 Mean Square Error (MSE)

Simulation was carried out to detect the average squared difference between the original data and the corrupted one. This is known as the MSE of the system. The error that occurs during transmission is the difference between the value of the original data and the corrupted data. In this paper, simulation was carried out to detect the effect of the turbo code when applied in the transmission and when turbo code was not. Simulation was carried out to determine the MSE of the reconstructed video using the algorithm shown in equation 7

$$MSE = i/n\sum_{i=0}^{n-1}(I-i)K_i \qquad (7)$$

I denotes the numerical quantity of the original video, and K(i) is the numerical quantity of the reconstructed video.

The simulation was carried out using 400bits frame size at sixth decoding iteration with a punctured code rate ½. The result of the simulation was shown in Figure 10, where the MSE of the reconstructed video using a turbo code system was much lower than when turbo code was not applied. It is shown that, when the SNR is 0.5dB the MSE of the reconstructed video with Turbo code was 39.20 but without turbo code was 45.78. when the SNR changes to 1dB the MSE result with turbo code was 36.59 while without turbo code was 44.14. It can be seen that as the SNR ranges increases the MSE value reduces. The lower the MSE value the better the reconstructed video. That is, the video has recovered from error and is well suitable for a proper diagnosis of the patient by the medical expert.

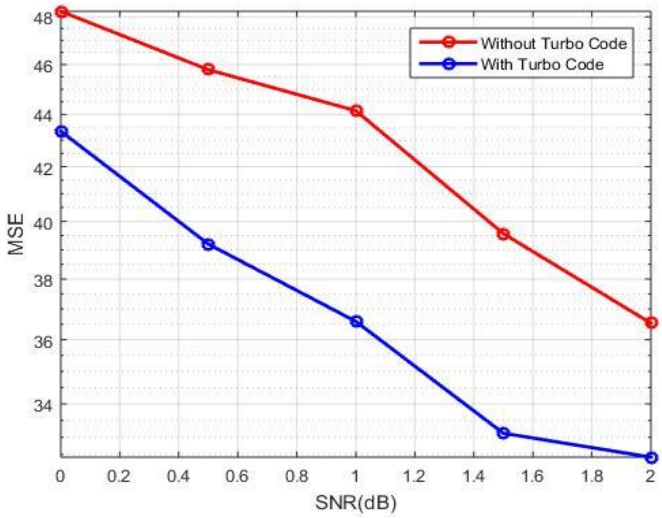


Figure 10: The MSE of the reconstructed video when turbo code was applied and without Turbo code at varied SNR ranges with 400bits frame size at the sixth decoding iteration using punctured code rate 1/2.

5.4 Processing Time of the Decoders

Simulation was carried out to determine the computational efficiency of the decoders used in the turbo code system. The input data frame size was varied from 50bits to 400bits and the number of decoding iteration was 6, with the decoding code rate of ½ at varied SNR level. An iterative decoding process took place between the decoders to check the number of errors after each decoding iteration. When the number of errors is zero at a particular iteration, the decoder will not execute the next iteration so as decrease the processing load. In the simulation, a number of six decoding iteration was carried out to determine the Processing time of the decoders. The result of the simulation was shown in Figure 11.

From the figure, it can be seen that when the SNR is 1 and the frame size is 50bits the processing time is 20.45(s), when the frame size is150bits, the processing time is 21.54(s). The results sshows that the processing time of the decoders increases as the frame size and the SNR increases. Also simulation was carried out with frame size of 400bits at various level of SNR with decoding code rate ½ at the sixth iteration. The result of the simulation was shown in Figure 12. From the graph it can be seen that the processing time of the decoder is increasing as the SNR range increases. In other to maintain a lower processing time of the decoders, the ratio between the signals to noise ratio must be kept very low in other to achieve a good transmission system.

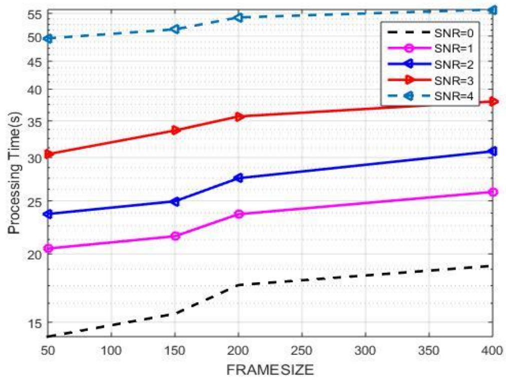


Figure 11: The Processing time of the decoder at different frame sizes at a varied range of SNR for six decoding iteration at code rate of 1/2 in AWGN channel.

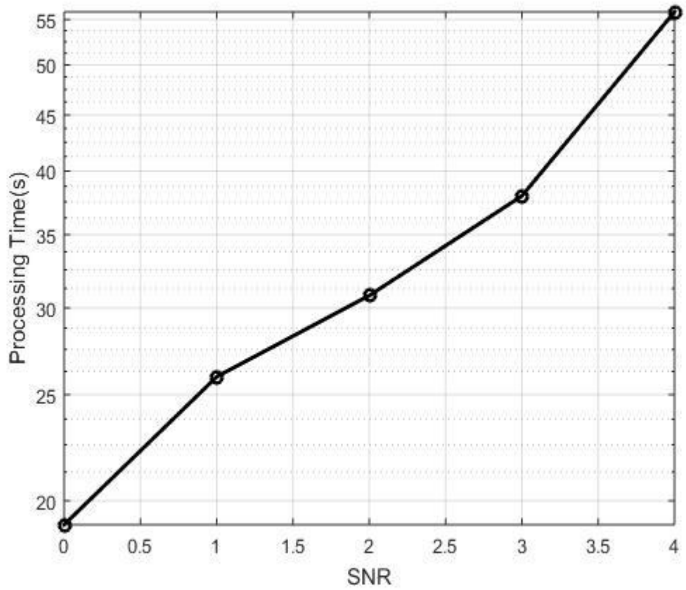


Figure 12: Processing time of the decoders at different SNR level with frame rate =400

6. CONCLUSION

The details of the developed system is given in this paper. The video from the patient site was first converted to a binary sequence before it was fed into the turbo encoder. Two Recursive Systematic Code (RSC) encoder was used and are punctured to have a code rate of 1/2. Then noise was introduced into the coded bits. The noise was added by activating the AWGN channel. The coded sequence received from the channel are fed into the turbo decoder and are implemented using SOVA decoding Algorithm. The error in the video was detected and corrected after passing through a number iteration before it was sent to the medical expert for proper diagnosis.

The effect and performance of various parameters on the performance of turbo coding such as number of decoding iteration, frame sizes, puncturing were investigated and analyzed. Based on the result of the simulation, the performance of turbo code for error detection and correction can be improved by reconfiguring the different parameters of the turbo code to achieve a better BER. Therefore, turbo code can be reconfigured dynamically to meet up with the resource prerequisite in the communication network.

When configuring Telemedicine application and other application that is sensitive to delay, the system should be set up with lower number of iterations, higher code rate and not to large frame sizes. But for applications that are not sensitive to delay the system can be configure using higher number of iteration, frame size and other parameters to achieve the desired BER. This makes turbo code to be suitable for variety of applications such as real time voice and video, compressed video playback, file and data transfer based on the availability of network infrastructure and quality of service required.

7. ACKNOWLEDGMENTS

Our appreciation to our God, the giver of life and the institution that provided us an enabling environment and support that contributed to the success of this work.

8. REFERENCES

Aderemi A. Atayero, Yury A. Ivanov (2013). Chapter 9 Modeling of Packet Streaming Services in Information Communication Networks", IGI Global.

A Ajala Funmilola, D Fenwa Olusayo, A. Aku Michael (2015). Comparative analysis between discrete cosine transform and wavelet transform techniques for medical image compression", International Conference on Computer Vision and Image Analysis Applications.

Dejan Dinevski, Robi Kelc, Bogdan Dugonik (2011). Chapter 10 Video Communication in Telemedicin, InTech.

Emuoyibofarhe O.J. and Bardi M.I. (2009). Simulation of a channel control scheme for error detection and correction in wireless communication systems. Afr. J. Comp. & ICT Vol. 2, No. 1, pp. 21 –29 ISSN 2006 – 1781.

Hannah Kirse (2014). Studies in Error Correction Coding. A Senior Thesis submitted in partial fulfillment of the requirements for graduation in the Honors Program.Liberty University Spring 2014.

Huyu Qu, Le Yi Wang, Christopher M. Klaus, Qiang Cheng, Ece Yaprak and Hong Wang (2011). Wireless Telemedicine System: An Accurate, Reliable and Secure Real-time Health Care, Telemedicine Techniques and Applications, Prof. Georgi Graschew (Ed.), ISBN: 978-953-307-354-5, InTech.

Kachienga, Michael Ogembo. (2011). Challenges in managing diffusion of telemedicine technology in South Africa", First International Technology Management Conference.

Lathi, B. P. and Ding, Z. (2009). Modern digital and analog communication systems (4th ed.). New York: Oxford University Press.

Martini M. G and Mazzotti M. (2006). Quality driven wireless video transmission for medical. Applications. in *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS).*

Mishra, S.and Mishra, KC. (2006). Medical Informatics: An Exploration. pp 4, ICFAI Books,ICFAI University Press, ISBN: 81-314-0378-5, Hyderabad.

Ranjeet Singh Tomar. (2009). Performance Evaluation Techniques for Mobile Communication Systems. First International Conference on Computational Intelligence Communication Systems and Networks.

Seyed MohammadSadegh Dashti. (2017). Real-word error correction with trigrams: correcting multiple errors in a sentence, Language Resources and Evaluation.

Stallings, W. (2006). Data and computer communications. Upper Saddle River: Prentice Hall.

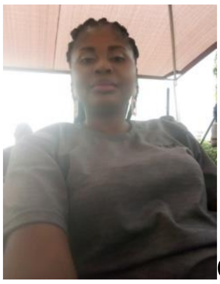
Shannon limit. (2001). Retrieved from Hargrave's Communications Dictionary.

Wah B. W., Su X., and Lin D. (2000). A Survey of Error-Concealment Schemes for Real-Time Audio and Video Transmissions over the Internet," presented at Proceedings. IEEE International Symposium on Multimedia Software Engineering.

Wang (2013). TCP/IP and Interne, Computer Architecture and Security Fundamentals of Designing Secure Computer Systems.

Wang Y. and Zhu Q.-F.(1998). Error Control and Concealment for Video Communication: A Review. *Proceedings of the IEEE*, vol. 86, pp. 974 – 997.

Author's Brief Profile



Olayinka Oyeronke O. holds a Bachelor of Technology (B.Tech) degree in Computer Engineering and Master of Technology (M.Tech) degree in Computer Science both from Ladoke Akintola University of Technology, Ogbomoso, Oyo State, Nigeria. Her areas of research includes Computer and Telecommunication Networks, e-Health, Telemedicine. She can be reached on +2348057933338 and olayinkaoyeronke@gmail.com.

Emuoyibofarhe .O. Justice (PhD) is currently a Professor in the Department of Information System at Ladoke Akintola University of Technology, Ogbomoso, Oyo State, Nigeria. His research areas include Computational Optimization, Neural Networks, Mobile Computing,

Wireless Communication, E-Health and Telemedicine. He can be reached on +2348033850075 and eojustice@gmail.com

Oladosu John B. (PhD) is currently an Associate Professor in the Department of Computer Engineering at Ladoke Akintola University of Technology, Ogbomoso, Oyo State, Nigeria. His research areas include Satellite Systems and Smart Antennas, computer and Telecommunications Networks, A.I., Expert Systems and Soft Computing Applications. He can be reached on +2348034556065 and johnoladosu@gmail.com.