International Journal of Computer Applications (0975 – 8887)
Volume 50 – No.1, July 2012

Analysis of Video Coding and Error Resilience Tools of H.264/AVC in Wireless Environment

Wajahat Javed,
Aamir Khan
COMSATS Institute of Information Technology, WahCantt

Khurram Shahzad,
Muhammad Asif,
COMSATS Institute of Information Technology, WahCantt

Faisal Munir
COMSATS Institute of Information Technology, WahCantt

ABSTRACT
Recently more and more telecommunication systems are supporting different kinds of real-time transmission, video transmission being one of the most important application.In wireless environments, channel bandwidth and high packet loss rate are to main limitations in the way of delivering of a good quality video to the end user. Therefore, in applications such as video over wireless networks, a video codec should have ability to handle the erroneous situations of the channels well as the bandwidth limitations.H.264/AVC is the newest international video coding standard, jointly developed by groups from ISO/IEC and ITUT.It has several error resilience techniques to make a video bit stream robust in the erroneous channels conditions and also achieves a significant improvement in the compression efficiency. We analyze various error resilienceschemes and innovative featuresof H.264/AVC for real time video streaming .The focus of the work is to test video coding and error resilience tools of H.264/AVC in real timeenvironment over wireless networks.

1. INTRODUCTION
This paper is about a real time Video Streaming over Wireless Network. The main goal of this project is to test video coding and error resilience tools of H.264/AVC in real time over wirelessmedium. Model of the project is shown as in figure below.

Fig. 1 Modal

The size of raw video is big and is not feasible to transmit this format over any wired and wireless network, so this format needs to be compressed. Without compression it is not possible to transmitvideo without affecting the quality over wireless network. The main issue comes without compression is low bandwidth and high bit rate, because in low bandwidth it is unable to send a big file. So to transmit video over wireless network first the video must be compressed and then it is able to send transmit real time video over wireless network without effecting the quality. The new and latest compression technique H.264 (AVC) applies on video for best compression.

The main objective is to transmit real time video over wireless network without affecting the quality. The video must be compressed by using H.264 Advanced Video Codec (AVC). Then this compressed video is transmitting over wireless network and observes the performance ofthe coding and error resilience tools of H.264/AVC in real time environments.

In daily life use of high-quality and low-bit rate video communication is growing day by day.The ending of the 20th century multimedia traffic, especially which video streaming has increased by using H.264/Advanced Video Coding (AVC) standard. The exceptional performance of H.264/AVC has attracted the attention of the electronics industry, and some broadcasted digital programs on wireless mobile phones use a video codec made with the H.264/AVC standard. Streaming video is becoming famous over current generation of mobile wireless network. The applications expected for the H.264/AVC standard consist of broadcast over cable, multi-media streaming services over cable modem, local area network (LAN),and wireless networks; conversational services over wireless and mobile networks, and multimedia messaging services over LAN, wireless, and mobile networks. With such wide application coverage, H.264/AVC quickly received a great deal of recent attention from industry and found wide-spread standard system.

2. RELATED WORK AND METHODOLOGY
Opencv (Open Source Computer vision) Library is used for getting live video from webcam. Opencv is an open source computer vision library. The library runs on Mac OS X, Windows and Linux.

It is necessary to configure Open CV library with the Operating system which should be used for getting video. After getting the video then the frame could be extracted from this video.Then these frames encodedusing H.264/AVC encoder on one node and after encoding the encoded file transmitted using TCP protocol on wireless peer to peer network. On the other node we receive the file and decode it using H.264/AVC decoder.

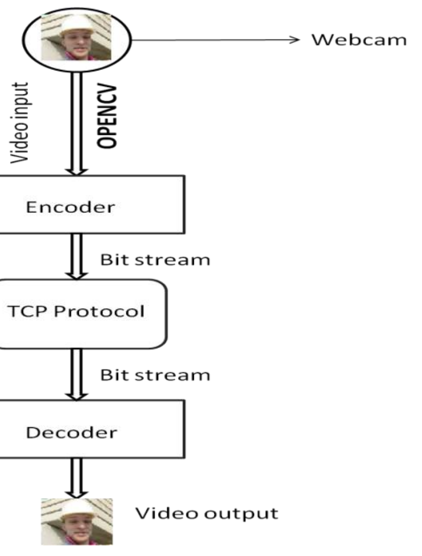


Fig. 2 Block Diagram

2.1. Encoder processes

Prediction:The H.264/AVC prediction methods are more flexible as compared to previous standards, enabling accurate predictions and hence efficient video compression. Intra prediction uses 16 × 16 and 4 × 4 block sizes to predict the macro block as we discussed earlier.Inter prediction uses differentblock sizes from 16 × 16 down to 4 × 4 to predict pixels as explain in earlier variable block size motion estimation. The encoder also gives flexibility in the order of coding frames coded frames may occur before or after the current frame in display order.

Transform and Quantization:A residual macro block of frame is transformed using a 4 × 4 or 8 × 8 integer transform. It is the approximate form of DCT (discrete cosine transforms). The output of the transform coefficient isthen quantizedin quantization each coefficient is divided by an integer value. Quantization reduces the precision of the transform coefficients according to a quantization parameter (QP). For example, the original coefficientvalues in Figure 4.12 are divided by a QP or step size of 8 and rounded to the nearest integer.Typically, the result is a block in which most or all of thecoefficients are zero, with a fewnon-zero coefficients. Setting QP to a greater value means, greater compression by more coefficients set to zero. It may lead to poor video quality. Setting QP to lower value means that lower compression of video by setting non-zero quantization.

Encoder Parameters:In the encoder module Input sequence is YUV format video with frame rate 30 Frame Rate per second. This is the video we getting directly for webcam. In encoder baseline profile is with 4x4 transform and CAVLC (Context-based Adaptive Variable Length Coding) Entropy coding method is used. To minimize computational complexity bi-directional B frames are not used and only I frame and P frames are used i.e. IPPP. 2 previous reference frames used for inter motion search. The GOP length of 10 frames is used.

Slicing is introduced with fixed number of macro block. 3,9,11 slices are used for experimenting. Encoder is also set for FMO with two slice group with checkerboard /dispersed pattern. FMO2, FMO3, FMO4, FMO5 is set for experimentation.[9]

The encoder is set for output of RTP (Rapid Transport Protocol) packet file format with CBR (constant bit rate). Bit rate is 128Kbps having initial quantization parameter 28 for 1st I frame.

2.2. Server

For transmitting video from one node to other we use windows socket programming. Using windows socket programming, socket is created. An address is assigned to server which is its own address. The server socket address is bind with a server's well known port. W. The socket is converted into listening socket. The server waits for client connection to complete. Three way hand shack between server and client is made. The file we want to transmit, we read it in binary form makepackets and these packets send server to client. The protocol which we use is TCP protocol. TCP offers a byte-oriented, assured transport service, which is based on re-transmission and timeout mechanisms for error control.

2.3. Client

For receiving a video to other node we make a client using again windows socket programming. Using windows socket programming on client end socket is created, the server address is assigned to client(because client must know the address of destination) with a port address from where client connect and receive file. Client receives the encoded file from server and writes this coded file to the path which we set. The whole process of server and client is shown in fig 3.

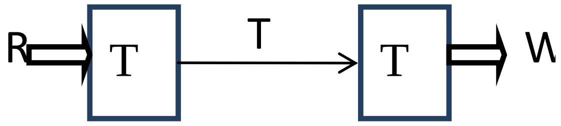


Fig. 3 Server client process

2.4. Decoder

The decoder is complement of encoder. The video decoder reconstructs a video frame from the compressed bit stream. The received coded bit stream is decoded with H.264 decoder. In decoder module each value is set as it was set in encoder module**.**

The decoder hasfollowing processes

Bit stream decoding:A video decoder receives the compressed H.264 bit stream and decodes each of the syntax elements and extracts the information of quantized transform coefficients, predictioninformation, etc. This information is then used to reverse the coding process and reconstructsequences video frames.

Inverse Quantization and inverse transform:Inverse quantization is inverse process of quantization. In inverse quantization every coefficient is multiplied by an integer value to restore its original scale. The original values of coefficients as they are in encoder side are not restorable atdecoderside because quantization is lossy a procedure. As shown in fig 3.4 the quantized coefficient is multiplied by a QP value of 8 for rescaling of transform coefficients at decoder side. After inverse quantization process the inverse transform and inverse entropy processes are applied for reconstruct the approximated image.

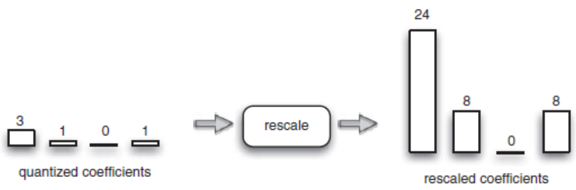


Fig. 4Rescaling

3. RESULTS

To evaluate the performance of error resilience shemes of H.264/AVC in wireless Dr.Altaf and Dr.Nadia[9] perform experiments for conversational video application. The B slices are not used in experimants. The experimental conditions are listed in followng table .The other parameters are that baseline used with frame type structure of an I frame followed by all P frames, i.e. ippp…….

Context Adaptive Variable Length Codes (dynamic Huffman entropic coding) is used in Baseline profile. B-frames are not used cause of computational complexity which increases the latency. Each slice has a single H.264 NAL packet. Thepacket loss rate used up to 50 %.

Table 1

Name	Size	Sampling
Foreman	QCIF(176 x144)	4:2:0
Bridge	QCIF(176 x144)	4:2:0

with above given experiment conditions the performance of differnent error resilience schemes evaluated at different packet loss rates[9]. As shown in figure 4.1(a) and 4.1(b)

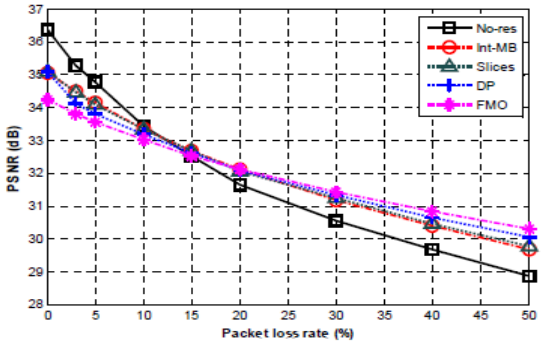
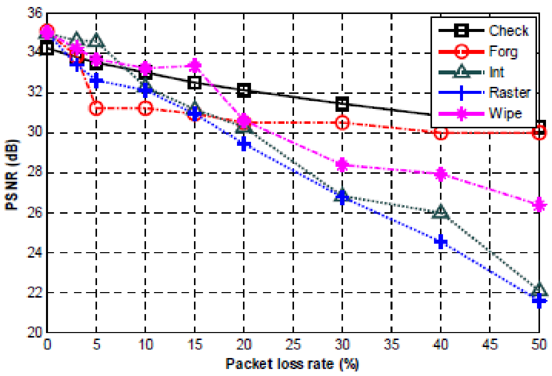


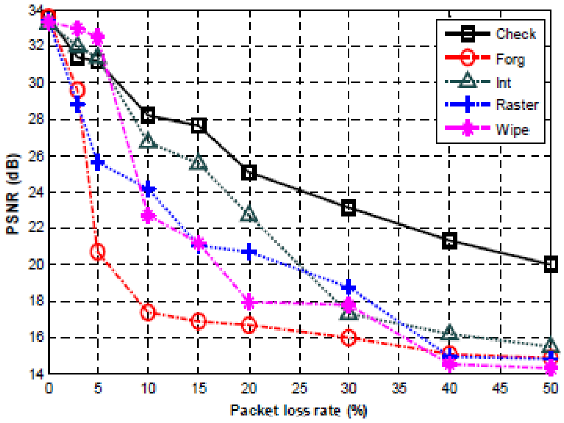
Fig 4.1Compression between several H.264/AVC error resilience methods and no resilience (n res) with isolated errors (a) for foreman and (b) bridege (closed)



o-

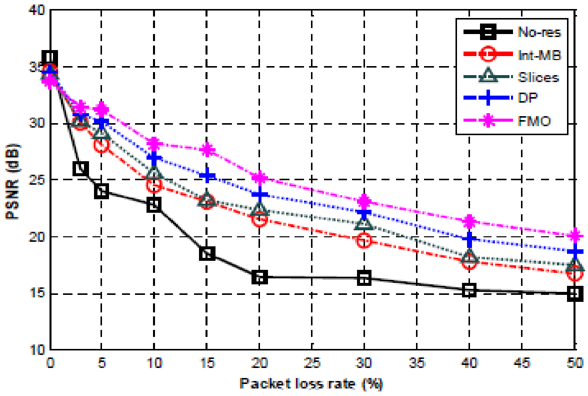
The FMO with different patterens also tested at packet loss rate up 50 % as shown in figure 4.2

(a)



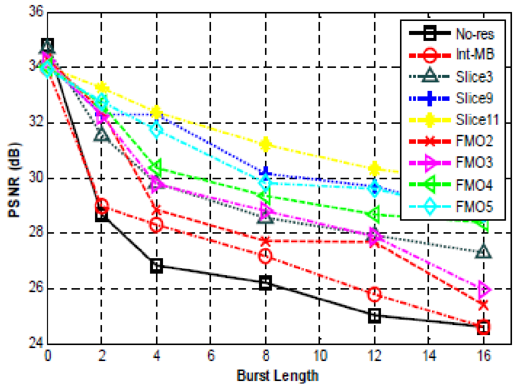
(b)

Fig. 4.2 The resulting video quality from five different FMO patterns within H.264 with uniform errors for (a) Foreman and (b) bridge (closed)[9]

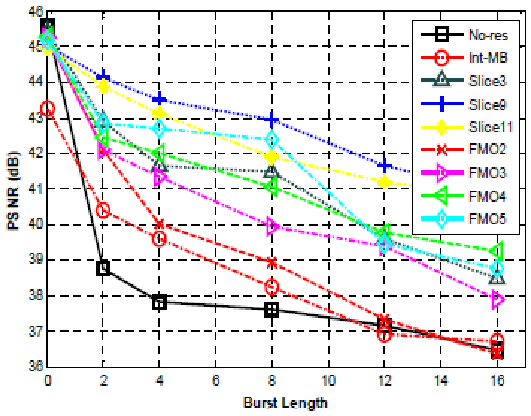


The experiment shows that at higer packet loss rates FMO checkerboard give better-quality as compared to other FMO patterns. For foreman at 10% the quality is below then 30 db. But for Bridge the quality is good up to 15% packet loss rate. From experiments [9] we observe that for Bridge sequence FMO checker board give the acceptable quality up to 40% loss rate.

The performance of error resilience schemes in brust errors for wireless enviorments also evaluated[9]. The experiments perform with same test conditions as explain above except frame rate was 10 Hz.



(Forman)



(Akiyo)

Fig. 4.3 Video quailty (PSNR) depening on frame brust length[9]

In these test conditions experiments shows that the graeter number of slices gives better quality as compared to FMO[9]. These experiments also shows that the packet length also decide the packet loss probability [9]. The smaller packet has less probability of loss.

M.M Gandhi and M.Ghanibari [12] also evluate performance of slice structuring with different number of slices per frame with different bit-error rate as shown in Fig 4.4. The slice structuring gives better quality as compared to with out slice structuring.But overhaed increase with increasing in slices per frame which degarde the quality in error free enviorments but give better quality in error enviorments [9].

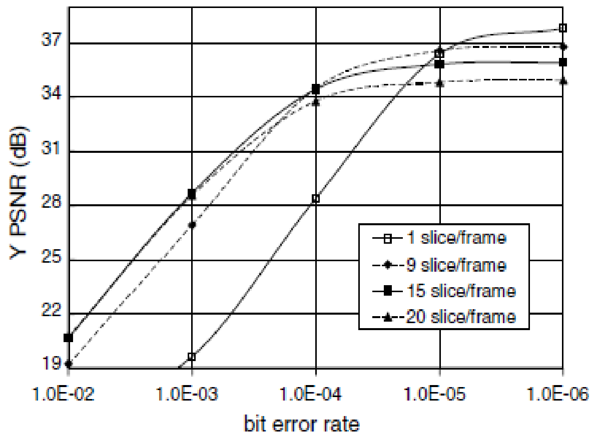


Fig 4.4DP enabled, with different slice/frame values

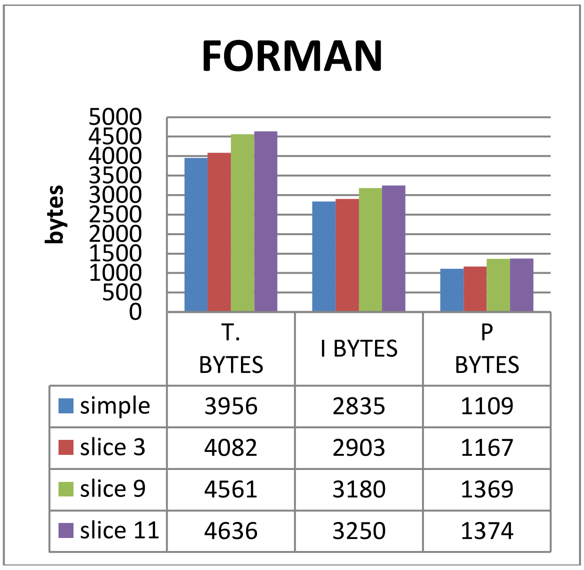
Altouh the slice structuring writh FMO and without FMO gives vary good video quality results in wireless enviornments sepecialy for mobile wireless enviornments where some time brust of errors take place.But the thing on wich we compromize for achiving good quality in wireless enviornments is data over head. The slice structuring introduced the data overhead as we increase the slices per frame . The increased overhead due to decreasing the slice size also effect on the coding performance [6].

The bussiness model in emerging wireless systems in which the end-user's costs are propotional to the transmitted data volume[6] and especially in mobile wireless systems where the transmitted data volume directly proportional to the cost. The data overhead directly effects on the cost.So we examined that how much data-overhaed increase with the incereasing of slices per frame and try to achieve compromize between the quality of video and data overhead which directly effect on the cost.

For experiments The foreman and akiyo clips was encoded QCIF video resolution (176x144pixel/frame) with 4:2:0 sampling, encoded with the reference JM 14 software for H.264/AVC.

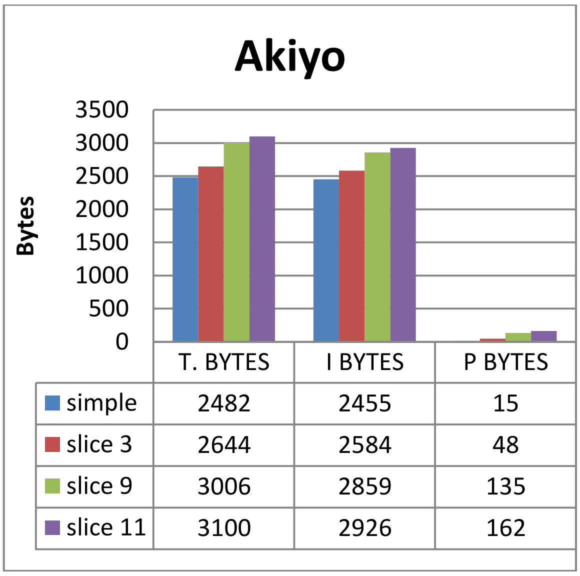
The parameters are that Baseline profile of H.264/AVC was selected with the frame type structure of an I-frame followed by all P frames, i.e. IPPP…..

The B frames are not used in baseline cause of computational complexity and also to reduce the latency. The encoder was set to output at constant Bit-Rate (CBR) at 128 kbps



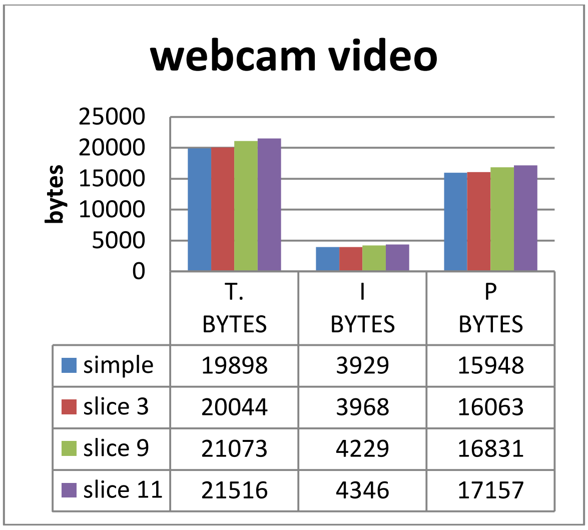
	T. BYTES	I BYTES	P BYTES
simple	3956	2835	1109
slice 3	4082	2903	1167
slice 9	4561	3180	1369
slice 11	4636	3250	1374

(a)



	T. BYTES	I BYTES	P BYTES
simple	2482	2455	15
slice 3	2644	2584	48
slice 9	3006	2859	135
slice 11	3100	2926	162

(b)



(c)

Fig 4.5The resulting data rate from different slice patterns

From the results in fig 4.5 (a) and (b) it is apparent that data overhead increase with increasing number of slices. The result shows for foreman that if we use 3 slices per frame then 2.4% data overhead increase on every I frame and 3.2% on total data.For 9 slices per frame 12 % data overhead increases on every I frame and 15 % on total data. For 11 slices per frame 14 % data overhead increases on every I frame and 17 % on total data. The akiyo video clip shows the same behaviour as the discuss earlier that data overhead sigficantly inrease with increase in number of slices. The increase of data overhead are shown in table 1.

The experiments also performs for raw video which we get from webcam with same experiment conditions as expalin above for foreman and akiyo video clip. The results are shown in table 2,3

These result shows same behaviour as shown by foremen and akiyo video clip but The rate of increase of data overhead vary from one video clip to other because of scene varitions.

Table 1 Forman overhead

Table 2 Akiyo overhead

Type	T. BYTES (%)	I BYTES (%)	P BYTES (%)
slice 3	6.5	5.2	220
slice 9	21	16.5	800
slice 11	25	19	980

Table 3 Webcam video overhead

Type	T. BYTES (%)	I BYTES (%)	P BYTES (%)
slice 3	0.73	0.99	0.72
slice 9	5.9	7.6	5.6
slice 11	8.1	10.6	7.5

The FMO results shows that the FMO with one slice group and with two slice group have the same encoding bytes for both akiyo and foreman but data encoding bytes are increase if further slicing introduce in frame as show in table 4,5,6.

The reults of video which we get from webcam (size of 640 x 480) using FMO with one slice group and with two slice groups have same results as the foreman and akiyo. No data overhead occers with FMO two slice group but as we inrease the slices per frame the data overhead increase.

The results of all experiments shows same behaviour for slice structuring but the rate of inrease of data overhead cause of slice structuring vari from one frame to other cause of variations in scene, some scene are more complex then other cause of moving objects.

The data over head increase significantly with increasing number of siles which is not acceptable in mobile network enviorments where transmitted data is propotional to cost.

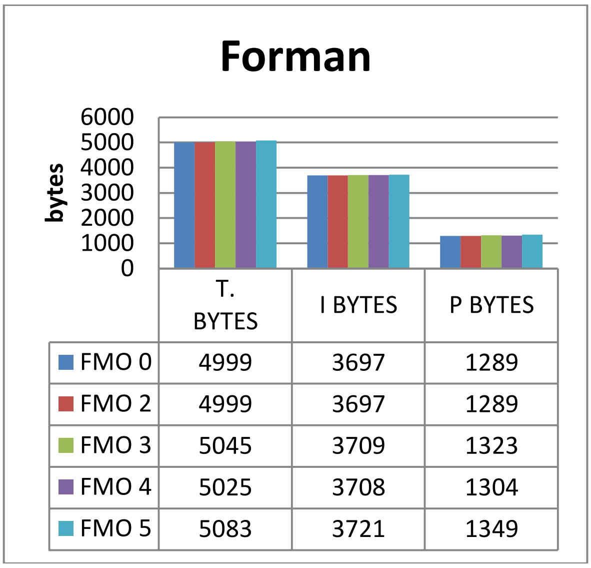
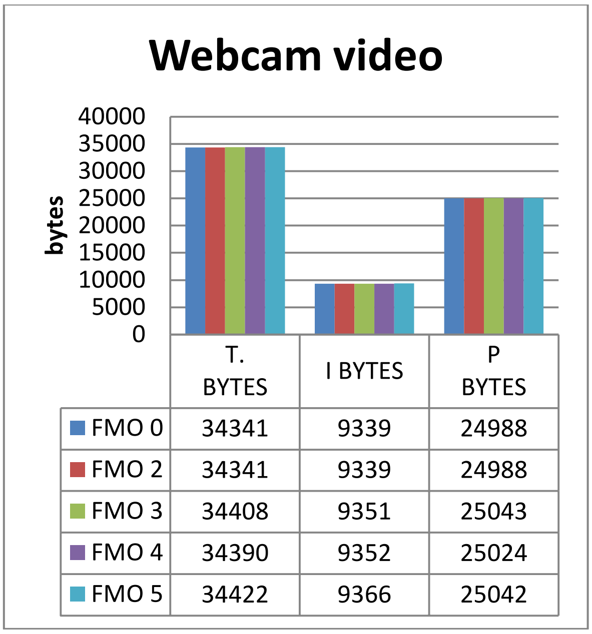


Table 4 Forman fmo overhead

Type	T.BYTES(%)	I BYTES (%)	P BYTES(%)
slice 3	3.2	2.4	5.3
slice 9	15.3	12.2	23.5
slice 11	17.2	14.6	23.9

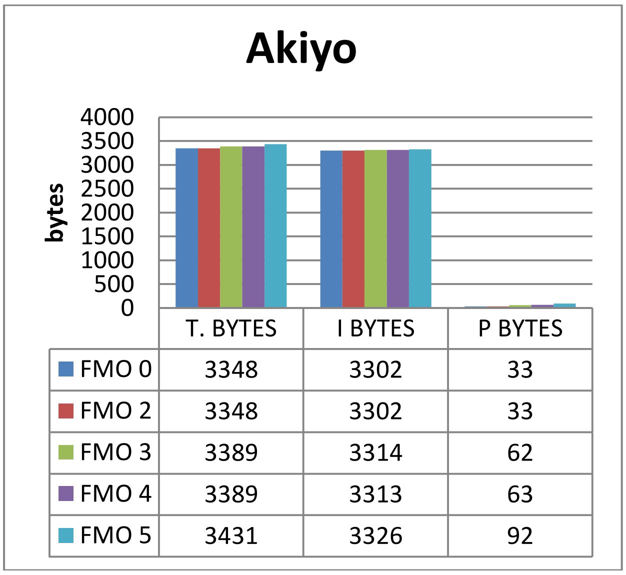
	T. BYTES (%)	I BYTES (%)	P BYTES (%)
FMO 2	0	0	0
FMO 3	0.92	0.32	2.6
FMO 4	0.52	0.32	1.2
FMO 5	1.7	0.65	4.7



	T. BYTES	I BYTES	P BYTES
FMO 0	34341	9339	24988
FMO 2	34341	9339	24988
FMO 3	34408	9351	25043
FMO 4	34390	9352	25024
FMO 5	34422	9366	25042

Table 5 Webcam video fmo overhead

	T. BYTES (%)	I BYTES (%)	P BYTES (%)
FMO 2	0	0	0
FMO 3	0.92	0.32	2.6
FMO 4	0.52	0.32	1.2
FMO 5	1.7	0.65	4.7



	T. BYTES	I BYTES	P BYTES
FMO 0	3348	3302	33
FMO 2	3348	3302	33
FMO 3	3389	3314	62
FMO 4	3389	3313	63
FMO 5	3431	3326	92

Proposed startegy

We observe from [9] results that with slices 3 gives acceptable quality up to 8 % packet loss rate for mobile users as shown in Fig 4.6 .W ith respect of good video quality slice 9 scheme proposed [9] but with this scheme the only 2 to 2.5 db video quality improves up to 8 % paket loss rate as compared to 3 slices per frame but minimum data overhead increase is 12% on every I frame as shown in table 1,2,3.So we proposed slice structuring scheme with 3 slices per frame with this scheme the data overhead much less then as compared to 9 slice scheme for mobile wireless enviornments as shown in table 1,2 and 3.

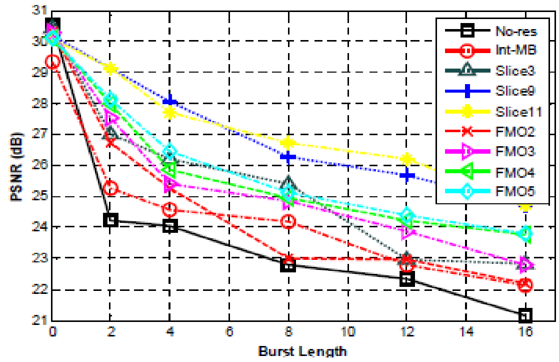


Fig 4.6 video quality (PSNR) depending on frame rust length for mobile [9]

4.CONCLUSION

The H.264/AVC video coding standard provide improved compression performance and a network-friendly design for conversational, storage, and streaming services .we explore the error resilience and innovative features and limmitations of error resiliencetechniques in real time video conversational services . Some experimentalresults were presented to show the performance and limitations of these error resiliency schemes. We conclude that video transmission over wireless network required the best compression technique and error resilience tecnique with minimum data overhead for delivering better quality video to the end user.

For conversational video services on wireless networks especially in mobile network the selection of error resilience technique is very sensitive issue. Slice structuring with 3 slices per frameis best compromise between quality and data over head in erroneous channels especially in mobile networks.

5. FUTURE WORK

The H.264/AVC has more options and parameters and cover wide range of applications as compered to any previous standerd codec . The performance of H.264/AVC is highly depend on the selection of tools and parameters acording to application.Right selection gives high compression performance and wrong selection gives poor quality. Many researchers around the world research on the performane of error resilence tools of H.264/AVC in wired and wireless networks with respect of application but these results are simulations based. The future work is that extract slices from H.264/AVC encoded file , transmit these slices (one slice as one NAL packet) in different real time wireless enviornments. Then observe the performane of these tools of H.264/AVC in real enviornment.

Table 6 Akiyo fmo overhead

	T. BYTES (%)	I BYTES (%)	P BYTES (%)
FMO 2	0	0	0
FMO 3	1.2	0.3	88
FMO 4	1.2	0.3	88
FMO 5	2.4	0.7	178

6. REFERENCES

[1] H. Enomoto and K. Shibata, "Features of Hadamard transformed television signal," presented at the Nat. Conf. IECE Jpn., 1965, Paper881.

[2] H. C. Andrews and W. K. Pratt, "Fourier transform coding of images," in *Proc. Hawaii Int. Conf. System Sciences*, 1968, pp.677–679.

[3] N. Ahmed, T. Natarajan, and K. R. Rao, "On image processing and a discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1,pp. 90–93, Jan. 1974.

[4] DetlevMarpe and Thomas Wiegand, Heinrich Hertz Institute (HHI),Gary J. Sullivan, Microsoft Corporation"The H.264/MPEG4 Advanced Video Coding Standard and its Applications".

[5] Sunil Kumar ,LiyangXu a, Mrinal K. Mandalb,SethuramanPanchanathan 'Error resiliency schemes in H.264/AVC standard'

[6] ThomasWiegand, Gary J. Sullivan, *Senior Member, IEEE*, GisleBjøntegaard, and Ajay Luthra'Overview of the H.264/AVC Video Coding Standard'

[7] Jian-Wen Chen Chao-Yang Kao Youn-Long Lin 'Introduction to H.264 Advanced Video Coding'

[8] "THE H.264 ADVANCED VIDEO COMPRESSION STANDARD " byIain E.Richardson

[9] M. Altaf, M. Fleury , M. Ghanbari and N. Qadri "Error Resilience Performance for Wireless Conversational Servises", ACM 5th International Mobile Multimedia Communications.

[10] Stephan Wenger "H.264/AVC over IP"IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 13, NO. 7, JULY 2003

[11] A. Habibi, "Hybrid coding of pictorial data," IEEE Trans. Commun., vol. COM-22, no. 5, pp. 614–624, May 1974.

[12] M.M. Ghandi *, M. Ghanbari 'Layered H.264 video transmission with hierarchical QAM'

[13] "Advanced video coding for generic audiovisual services," Int.Telecommun. Union-Telecommun. (ITU-T) and Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC) JTC 1, RecommendationH.264 and ISO/IEC 14 496-10 (MPEG-4) AVC, 2003.

[14] "Video Compression—From Concepts to theH.264/AVC Standard"GARY J. SULLIVAN, SENIOR MEMBER, IEEE AND THOMAS WIEGAND